

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re U.S. Patent Application of)
MORI KI et al.)
Application Number: To be Assigned)
Filed: Concurrently Herewith)
For: SYSTEM AND METHOD FOR RESOURCE)
ACCOUNTING ON COMPUTER NETWORK)
ATTORNEY DOCKET NO. NITT.0170)

Honorable Assistant Commissioner
for Patents
Washington, D.C. 20231

**REQUEST FOR PRIORITY
UNDER 35 U.S.C. § 119
AND THE INTERNATIONAL CONVENTION**

Sir:

In the matter of the above-captioned application for a United States patent, notice is hereby given that the Applicant claims the priority date of July 8, 2003, the filing date of the corresponding Japanese patent application 2003-193335.

A certified copy of Japanese patent application 2003-193335 is being submitted herewith. Acknowledgment of receipt of the certified copy is respectfully requested in due course.

Respectfully submitted,

Stanley P. Fisher
Registration Number 24,344

REED SMITH LLP
3110 Fairview Park Drive
Suite 1400
Falls Church, Virginia 22042
(703) 641-4200
January 6, 2004



Juan Carlos A. Marquez
Registration Number 34,072

PATENT OFFICE
JAPANESE GOVERNMENT

This is to certify that the annexed is a true copy of the following application as filed with this office.

Date of Application : July 8, 2003
Application Number : Patent Application No. 2003-193335
Applicant (s) : Hitachi, Ltd.

Dated this 27th day of November, 2003

Yasuo IMAI
Commissioner,
Patent Office

Certificate No. 2003-3098234



日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日 2 0 0 3 年 7 月 8 日
Date of Application:

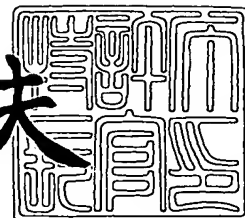
出 願 番 号 特 願 2 0 0 3 - 1 9 3 3 3 5
Application Number:
[ST. 10/C]: [J P 2 0 0 3 - 1 9 3 3 3 5]

出 願 人 株式会社日立製作所
Applicant(s):

2 0 0 3 年 1 1 月 2 7 日

特許庁長官
Commissioner,
Japan Patent Office

今 井 康 夫





【書類名】 特許願

【整理番号】 NT03P0533

【提出日】 平成15年 7月 8日

【あて先】 特許庁長官 殿

【国際特許分類】 G06F 11/34
G06F 17/60
G06F 13/24

【発明者】

【住所又は居所】 東京都国分寺市東恋ヶ窪一丁目 2 8 0 番地 株式会社日立製作所中央研究所内

【氏名】 森木 俊臣

【発明者】

【住所又は居所】 東京都国分寺市東恋ヶ窪一丁目 2 8 0 番地 株式会社日立製作所中央研究所内

【氏名】 上原 敬太郎

【発明者】

【住所又は居所】 東京都国分寺市東恋ヶ窪一丁目 2 8 0 番地 株式会社日立製作所中央研究所内

【氏名】 對馬 雄次

【発明者】

【住所又は居所】 東京都国分寺市東恋ヶ窪一丁目 2 8 0 番地 株式会社日立製作所中央研究所内

【氏名】 田中 剛

【特許出願人】

【識別番号】 000005108

【氏名又は名称】 株式会社日立製作所



【代理人】

【識別番号】 100068504

【弁理士】

【氏名又は名称】 小川 勝男

【電話番号】 03-3661-0071

【選任した代理人】

【識別番号】 100086656

【弁理士】

【氏名又は名称】 田中 恭助

【電話番号】 03-3661-0071

【選任した代理人】

【識別番号】 100094352

【弁理士】

【氏名又は名称】 佐々木 孝

【電話番号】 03-3661-0071

【手数料の表示】

【予納台帳番号】 081423

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要



【書類名】 明細書

【発明の名称】 サーバリソース集計方法、サーバリソース集計システム、および、そのためのサーバ

【特許請求の範囲】

【請求項 1】 計算機システムのサーバの使用リソースを集計するサーバリソース集計方法において、

この計算機システムは、前記サーバで使用されたサーバリソースを集計する集計サーバを有し、前記サーバと前記集計サーバとは、ネットワークで接続され、

前記サーバは、OS またはアプリケーションソフトウェアを動作させる CPU と、前記 CPU と処理を独立におこなう制御装置とを有するサーバであって、

前記サーバを使用するユーザの情報を登録するステップと、

前記登録されたユーザが、OS またはアプリケーションソフトウェアを動作させることにより、サーバのリソースを使用するステップと、

前記制御装置が、前記 CPU 上で動作している OS またはアプリケーションソフトウェアと通信するステップと、

前記制御装置が、前記通信によって、前記サーバリソースを使用したユーザのユーザ情報とそのユーザの使用したサーバリソース量と含む使用リソース情報を取得するステップと、

前記制御装置が、取得した前記使用リソース情報を前記集計サーバに送信するステップと、

前記集計サーバが前記使用リソース情報を受信し、前記ユーザごとに集計するステップとからなることを特徴とするサーバリソース集計方法。

【請求項 2】 前記ユーザの情報を登録するステップで、そのユーザをこの計算機システム内で一意に特定するグローバルユーザ ID を設定し、そのユーザの使用するサーバを、この計算機システム内で一意なサーバ ID によって指定し、そのユーザを、そのサーバ内で一意に特定するローカルユーザ ID を設定する処理を含み、

前記制御装置が、前記使用リソース情報を前記集計サーバに送信するステップで、前記ユーザを一意に特定する情報として、そのユーザに対応するローカルユ

ーザIDと、そのサーバに対応するサーバIDを送信する処理を含み、

前記集計サーバが前記使用リソース情報を受信し、前記ユーザごとに集計するステップで、受信した前記ローカルユーザIDと前記サーバIDとから、そのユーザに対応する前記グローバルユーザIDを特定して、前記特定されたグローバルユーザIDごとに前記使用されたサーバリソース量を集計する処理を含むことを特徴とする請求項1記載のサーバリソース集計方法。

【請求項3】 前記ローカルユーザIDとして、前記サーバ上で動作しているOSの管理するユーザIDを用いることを特徴とする請求項2記載のサーバリソース集計方法。

【請求項4】 前記OSまたはアプリケーションソフトウェアは、前記ユーザを認証し、そのユーザの要求に基づきスレッドの実行を開始し、そのスレッドごとによって使用されたサーバリソース情報を収集し、収集されたサーバリソース情報を、前記ローカルユーザIDごとに前記制御装置に送信することを特徴とする請求項2記載のサーバリソース集計方法。

【請求項5】 前記集計サーバが前記制御装置に対し、前記使用リソース情報の送信を要求するステップをさらに有することを特徴とする請求項1記載のサーバリソース集計方法。

【請求項6】 前記集計サーバが前記制御装置に対し、前記使用リソース情報の送信を要求するステップで、前記使用リソース情報の収集の対象とするユーザを指定する情報を、前記使用リソース情報の送信の要求に付加し、その送信の要求を前記ネットワークにブロードキャストによって送信する処理を含み、

前記サーバが前記使用リソース情報を前記集計サーバに送信するステップで、

前記サーバが前記使用リソース情報の収集の対象とするユーザの使用リソース情報を保持している場合には、そのユーザの使用リソース情報のみを選択して送信し、

前記サーバが前記使用リソース情報の収集の対象とするユーザの使用リソース情報を保持していない場合には、そのユーザの使用リソース情報を保持していないことを示す情報を送信する処理を含むことを特徴とする請求項5記載のサーバリソース集計方法。

【請求項 7】 前記サーバを使用するユーザの情報を登録するステップで、グラフィカルユーザインタフェース（G U I）を利用して登録することを特徴とする請求項 1 記載のサーバリソース集計方法。

【請求項 8】 計算機システムのサーバの使用リソースを集計するサーバリソース集計システムにおいて、

この計算機システムは、前記サーバで使用されたサーバリソースを集計する集計サーバを有し、前記サーバと前記集計サーバとは、ネットワークで接続され、

前記サーバは、O S またはアプリケーションソフトウェアを動作させる C P U と、前記 C P U と処理を独立におこなう制御装置とを有するサーバであって、

前記集計サーバは、前記サーバを使用するユーザを登録する手段を有し、

しかも、前記サーバは、前記登録されたユーザが、O S またはアプリケーションソフトウェアを動作させることにより使用したサーバリソース量を、前記ユーザごとに記録する手段を有し、

前記制御装置は、前記 C P U 上で動作している O S またはアプリケーションソフトウェアと通信し、その通信によって、前記サーバリソースを使用したユーザのユーザ情報とそのユーザの使用したサーバリソース量と含む使用リソース情報を取得する手段と、

取得した前記使用リソース情報を前記集計サーバに送信する手段とを有し、

前記集計サーバは、さらに、前記制御装置からの前記使用リソース情報を受信し、前記ユーザごとに集計する手段を有することを特徴とするサーバリソース集計システム。

【請求項 9】 前記 O S またはアプリケーションソフトウェアは、前記ユーザを認証するユーザ認証部と、そのユーザの要求に基づきスレッドの実行を開始するスレッド実行開始部と、そのスレッドごとによって使用されたサーバリソース情報を収集する収集手段と、収集されたサーバリソース情報を、前記制御装置に送信する送信手段を有することを特徴とする請求項 8 記載のサーバリソース集計システム。

【請求項 10】 前記集計サーバは、前記サーバを使用するユーザの情報を登録する際に、前記ネットワークを経由して登録することを特徴とする請求項 8

記載のサーバリソース集計システム。

【請求項 11】 使用リソースを収集する機能を有するサーバにおいて、
このサーバは、OS またはアプリケーションソフトウェアを動作させる CPU
と、前記 CPU と処理を独立におこなう制御装置とを有するサーバであって、
前記 OS またはアプリケーションソフトウェアを動作させることにより使用し
たサーバリソース量を、前記ユーザごとに記録し、前記サーバリソースを使用し
たユーザのユーザ情報とそのユーザの使用したサーバリソース量と含む使用リソ
ース情報として収集する手段を有し、

前記制御装置は、前記 CPU 上で動作している OS またはアプリケーションソ
フトウェアと通信する手段と、

その通信によって、前記使用リソース情報を取得する手段とを有することを特
徴とする使用リソースを収集する機能を有するサーバ。

【請求項 12】 前記制御装置は、前記使用リソース情報の取得を要求する
手段として、前記 CPU に対して割り込みを実施する割り込み回路を有し、

前記使用リソース情報を収集する手段として、前記割り込みにより起動される
デバイスドライバを有し、

そのデバイスドライバは、前記 OS の提供する API を呼び出して、前記使用
リソース情報を収集することを特徴とする請求項 11 記載の使用リソースを収集
する機能を有するサーバ。

【請求項 13】 前記制御装置は、前記使用リソース情報の取得を要求する
手段として、その制御装置から書き込み可能で、かつ、前記 CPU から読み出し
可能なレジスタと、そのレジスタに対して使用リソース情報の取得を要求する識
別子を書き込む手段とを有し、

前記使用リソース情報を収集する手段として、前記 OS またはアプリケーショ
ンソフトウェアが、前記レジスタをポーリングし前記識別子を書き込まれたこと
を検出する検出手段と、前記 OS またはアプリケーションソフトウェアの提供す
る API を呼び出す API 呼出部と、前記使用リソース情報を収集する収集部を
有することを特徴とする請求項 11 記載の使用リソースを収集する機能を有する
サーバ。

【請求項 14】 前記サーバは、さらに、主記憶装置を有し、

その主記憶装置は、前記制御装置から書き込み可能で、かつ、前記 CPU から読み出し可能な特定の領域を有し、

前記制御装置は、前記使用リソース情報の取得を要求する手段として、前記特定の領域に対して、使用リソース情報の取得を要求する識別子を書き込む手段を有し、

前記使用リソース情報を収集する手段として、前記 OS またはアプリケーションソフトウェアが、前記特定の領域をポーリングし、前記識別子を書き込まれたことを検出し、OS またはアプリケーションソフトウェアの提供する API を呼び出して、前記使用リソース情報を収集することを特徴とする請求項 11 記載の使用リソースを収集する機能を有するサーバ。

【請求項 15】 前記制御装置は、前記 CPU から書き込み可能な記憶装置と、その記憶装置に対する書き込みを検出する手段をさらに有し、

前記 OS またはアプリケーションが、前記使用リソース情報を前記記憶装置に書き込むことによって、前記制御装置が前記使用リソース情報を取得することを可能にすることを特徴とする請求項 11 記載の使用リソースを収集する機能を有するサーバ。

【請求項 16】 前記サーバは、さらに、主記憶装置を有し、

その主記憶装置は、前記 CPU から書き込み可能で、かつ、前記制御装置から読み出し可能な特定の領域を有し、

前記 OS またはアプリケーションが、前記使用リソース情報を前記特定の領域に書き込むことによって、前記制御装置が前記使用リソース情報を取得することを可能にすることを特徴とする請求項 11 記載の使用リソースを収集する機能を有するサーバ。

【請求項 17】 計算機システムのサーバの使用リソースに対する課金方法において、

この計算機システムは、前記サーバで使用されたサーバリソースを集計する集計サーバを有し、前記サーバと前記集計サーバとは、ネットワークで接続され、

前記サーバは、OS またはアプリケーションソフトウェアを動作させる CPU

と、前記CPUと処理を独立におこなう制御装置とを有するサーバであって、
前記サーバを使用するユーザの情報を登録するステップと、
前記登録されたユーザが、OSまたはアプリケーションソフトウェアを動作させることにより、サーバのリソースを使用するステップと、
前記制御装置が、前記CPU上で動作しているOSまたはアプリケーションソフトウェアと通信するステップと、
前記制御装置が、前記通信によって、前記サーバリソースを使用したユーザのユーザ情報とそのユーザの使用したサーバリソース量と含む使用リソース情報を取得するステップと、
前記制御装置が、取得した前記使用リソース情報を前記集計サーバに送信するステップと、
前記集計サーバが前記使用リソース情報を受信し、前記ユーザごとに集計するステップと、
前記ユーザごとに集計された使用リソース情報に基づいて、請求する金額を決定するステップと、
そのユーザに対して前記金額の請求をおこなうステップとからなるサーバの使用リソースに対する課金方法。

【請求項18】 前記制御装置が取得した前記使用リソース情報を前記集計サーバに送信するステップで、前記ユーザがサーバのリソースを使用した時刻情報を送信する処理を含み、

前記集計サーバがそのユーザに対して前記金額の請求をおこなうステップで、前記使用した時刻情報に基づいて生成したサーバリソース使用履歴を、そのユーザに送付する処理を含むことを特徴とする請求項17記載のサーバの使用リソースに対する課金方法。

【請求項19】 複数の論理区画に分割されたサーバにおいて、該サーバは各論理区画上で一つのOSと、0もしくは1以上のアプリケーションソフトウェアを稼働させるサーバであって、一つ以上のCPUと前記CPUと独立して処理をおこなう制御装置を有し、

前記制御装置は、一部もしくは全ての論理区画上のOSもしくはアプリケーシ

ョンソフトウェアに対して、サーバリソースを使用したユーザのユーザ情報と該ユーザの使用したサーバリソース量とを含む、使用リソース情報の取得を要求する手段を有し、

前記OSもしくはアプリケーションソフトウェアは、前記制御装置の要求にตอบสนองして使用リソース情報を返答する手段を有することを特徴とするサーバ。

【請求項20】 複数の論理区画に分割されたサーバにおいて、該サーバは、各論理区画上で一つのOSと、0もしくは1以上のアプリケーションソフトウェアを稼動させるサーバであって、一つ以上のCPUと前記CPUと独立して処理をおこなう制御装置と該制御装置からの要求を受付けるハイパバイザを有し、

前記制御装置は、前記ハイパバイザに対して、サーバリソースを使用したユーザのユーザ情報と該ユーザの使用したサーバリソース量とを含む、使用リソース情報の取得を要求する手段を有し、

前記ハイパバイザは前記制御装置の要求にตอบสนองし、一部もしくは全ての論理区画上のOSもしくはアプリケーションソフトウェアに対して該要求を転送する手段を有し、

前記OSもしくはアプリケーションソフトウェアは、該要求にตอบสนองして使用リソース情報を返答する手段を有することを特徴とするサーバ。

【請求項21】 請求項19または請求項20のサーバにおいて、前記使用リソース情報の取得を要求する手段として、割込み回路を有することを特徴とするサーバ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、サーバリソース集計方法に係り、サーバに負荷をかけず、かつ、既存のシステムにも容易に導入することのできるサーバリソース集計方法に関する。

【0002】

【従来の技術】

近年のインターネットの普及により、企業の情報システムの大規模化、複雑化

が進んでいる。これにより企業の所有するサーバの台数も飛躍的に増加し、企業のサーバ管理コストが増大している。これらの企業に代わりサーバの運用管理をおこなうアウトソーシングビジネスとして、IDC（インターネットデータセンタ、Internet Data Center）と呼ばれるビジネスが普及している。IDC事業者は、多数のサーバを運用管理し、そのサーバを多数の企業や団体などのユーザに貸し出して、日割りや月極めでサーバ使用料を徴収する。

【0003】

このIDCの一形態として、ユーティリティ・コンピューティングと呼ばれるビジネスモデルが注目を集めている。ユーティリティ・コンピューティングではIDC事業者が、ユーザの使用したCPU時間やメモリ容量、ストレージ容量、ネットワーク帯域などのサーバリソースに応じて使用料を徴収する。そのため、ユーザはあまりサーバを使用していない時期には、安い使用料しか払わなくてよいというメリットがある。このユーティリティ・コンピューティングにおいてはユーザごとに課金をおこなうため、多数のサーバで使用されたサーバリソースを集計し、ユーザ単位で正確に把握することが必要になる。

【0004】

サーバリソースの使用状況を把握する方法の従来技術としては、各サーバのOS上でエージェントと呼ばれるソフトウェアを動作させる方法が知られている（以下、第一の従来技術と呼ぶ）。

【0005】

この方法では、エージェントは定期的にシステムコール等によってOS (Operating System) と通信をおこない、サーバリソースの使用状況を収集する。次に、このエージェントは、集計サーバに収集したサーバリソースの使用状況を報告する。集計サーバは、その報告をユーザごとに集計し、サーバリソースの使用状況に応じてユーザに課金をおこなう。

【0006】

サーバリソースの使用状況を把握する二つ目の方法として、以下の特許文献1の「稼動記録保存型情報処理システム」が開示されている（以下、第二の従来技術と呼ぶ）。

【 0 0 0 7 】

この方法では、サーバに稼動状態検出手段を設け、稼動状態検出手段が主記憶に記録された性能コードと、CPUの稼動開始時刻、および、CPUの稼動終了時刻を組にして記録する。OSは、記録された性能コードに応じて、ユーザプロセスにCPU時間を割り付ける。課金者は、稼動状態検出手段に記録された情報（性能コードと、CPUの稼動開始時刻、および、CPUの稼動終了時刻）に基づいて課金をおこなう。

【 0 0 0 8 】

サーバリソースの使用状況を把握する三つ目の方法として、以下の特許文献2の「性能データ採取方式」が開示されている（以下、第三の従来技術と呼ぶ）。

【 0 0 0 9 】

この方法によるとOSが、非課金識別子によって課金を指示されているセグメント内のプロセスを実行中に更新されるプロセスタイマを備える。OSは、プロセスのロールアウト時に、プロセスタイマの使用している時間分は、非課金として、課金CPU時間から引くことにより課金CPU時間を算出する。課金者は、算出された使用CPU時間に基づき、課金をおこなう。

【 0 0 1 0 】

サーバリソースの使用状況を把握する四つ目の方法として、以下の特許文献3の「課金システム、課金制御回路、課金制御方法及び課金制御プログラムを記録した記録媒体」が開示されている。（以下、第四の従来技術と呼ぶ）

この方法では、サーバの拡張機器（ディスクドライブ装置やネットワーク機器など）に課金制御回路を設け、ユーザからのアクセス回数やアクセス時間を計測する。課金者は、計測されたアクセス回数やアクセス時間に基づき、課金をおこなう。

【 0 0 1 1 】**【特許文献1】**

特開平07-129271号公報

【特許文献2】

特開平06-095924号公報

【特許文献 3】

特開 2001-222336 号公報

【0012】

【発明が解決しようとする課題】

第一の従来技術では、各サーバの OS 上で動作するエージェントが、サーバリソースの使用状況を収集する。そのため、サーバの負荷が高くなった場合には、エージェント自体に CPU 時間などのサーバリソースが割り付けられなくなる。その場合、エージェントの動作が遅延し、一定の間隔でサーバリソースの使用状況を把握することが保証できなくなり、本来のサーバリソースの使用状況と異なる値が、エージェントから報告されるという問題がある。この問題を解決するために、エージェントをユーザのプロセスより高い優先度で実行する方法も考えられる。しかしその場合、エージェントで消費されるサーバリソースが増加し、ユーザのプロセスに十分なサーバリソースが供給されない恐れがあり、十分な解決とは考えられない。

【0013】

そのため、サーバの負荷が高くなっても、正確にサーバリソースの使用状況を把握することが解決すべき第一の課題である。

【0014】

第二の従来技術では、OS および稼動状態検出手段が主記憶に記録された性能コードを読み出す必要がある。この方法によると、OS は性能コードに従ってプロセスをスケジューリングすることが必須であり、この性能コードを考慮しない OS に対してはこの方法を適用することができない。また、稼動状態検出手段が性能コードを読み出す場合には物理メモリアドレスを指定することが必然であるが、OS の物理メモリ上のデータ配置はリビジョンによって異なる。そのためリビジョンの異なる OS に対しても適用することが出来ない。

一方、第三の従来技術では、OS のプロセスディスパッチャを変更することが必須であり、既存の OS に対して適用できない。また、Microsoft 社の Windows(R) のように、そもそもユーザごとの課金情報の収集を考慮していない OS に対しては適用できない。

【0015】

そのため、OSに依存せずにサーバリソースの使用状況を把握することが、解決すべき第二の課題である。

【0016】

第四の従来技術では、サーバリソースのうちCPU時間の使用状況を収集することができないという問題がある。この技術は、周辺機器に課金制御回路を設けるものだからである。また、CPUのアーキテクチャを変更して、この第四の従来技術を適用することを想定した場合、ユーザの膨大なソフトウェア資産との互換性を確保するために、テストやソフトウェアの移行作業などに大変な労力が掛かり、現実的でない。

【0017】

そのため、CPUを変更せずにCPU時間を含めたサーバリソースの使用状況を把握することが、解決すべき第三の課題である。

【0018】

本発明は、上記問題点を解決するためになされたもので、その目的は、サーバの負荷が高い場合であっても、正確にリソースの収集ができ、既存のOSやCPUのアーキテクチャでも変更することなく導入できるサーバリソース集計方法を提供することにある。

【0019】**【課題を解決するための手段】**

これらの課題を解決するために、本発明のサーバリソース集計システムでは、ユーザがリソースを使用するいくつかのサーバとそれらのサーバで使用されたサーバリソースを集計する集計サーバをネットワークで接続して構成する。

【0020】

そして、サーバは、OSまたはアプリケーションソフトウェアを動作させるCPUと、CPUと処理を独立におこなう制御装置とで構成するようにする。

【0021】

サーバでは、集計サーバから登録されたユーザが、OSまたはアプリケーションソフトウェアを動作させることにより、ユーザごとに使用したサーバリソース

量を監視している。

【0022】

そして、例えば、ある一定の時刻がたつと、制御装置よりOSのデバイスドライバに割り込みをかけて、OSにAPI呼び出しをさせることにより、そのユーザの使用リソース情報を収集させる。これは、OS上で動作しているアプリケーションソフトウェアが使用リソース情報を収集するようにしてもよい。

【0023】

OSまたはアプリケーションソフトウェアは、収集した使用リソース情報を、例えば、制御装置のメモリ上に書き込む。

【0024】

制御装置では、渡された使用リソース情報を集計サーバにネットワークを介して送信する。

【0025】

本方法では、使用されたサーバリソースの情報は、OS上で動作するエージェントではなく、CPUから独立した制御装置から、OSまたはアプリケーションソフトウェアに対して通信をおこなって取得される。一般に、OSは、割り込みなどのハードウェアからの要求を最優先に処理するため、OSの負荷が高くなっている場合であっても遅延なく使用されたサーバリソースの情報を収集できる。そのため、第一の課題を解決することができる。

【0026】

一般にハードウェアからの割り込みを受信するデバイスドライバは、OSごとに開発可能である。また、OSやアプリケーションから、サーバリソースの使用状況を取得するためのAPIは一般に開示されている。そのため、各OSに対応するデバイスドライバを開発することで本発明を適用可能であり、第二の課題を解決することができる。

【0027】

また、本発明では、サーバ上でCPUと別個に動作する制御装置を設けるため、CPUの変更をおこなうことなく、CPUのCPU時間を含めたサーバリソースの使用状況を把握することができ、第三の課題を解決することができる。

【0028】**【発明の実施の形態】**

以下、本発明に係る各実施形態を、図1ないし図20を用いて説明する。

【0029】**〔実施形態1〕**

以下、本発明に係る第一の実施形態を、図1ないし図14を用いて説明する。

(I) サーバリソース集計システムの構成

以下、図1ないし図4を用いて本実施形態に係るサーバリソース集計システムの構成について説明する。

【0030】

先ず、図1を用いて本実施形態における計算機システム全体の構成について説明する。

図1は、本発明の第一の実施形態における計算機システム全体の構成図である。

【0031】

サーバ101a、101b、101cは、管理用ネットワーク102とサービス用ネットワーク110に接続されている。サーバ101a、101b、101cは、ユーザからの要求に応じて処理を実行し、ユーザにはこれらのサーバで使用したCPU時間やメモリ容量などサーバリソースの数量に応じてサーバの使用料が課金される。ここでいうサーバとは、例えば、Webサーバ、メールサーバ、ファイルサーバ、データベースサーバなどである。

【0032】

管理用ネットワーク102には、サーバ101a、101b、101cの他に集計サーバ103、コンソール端末104、プリンタ105、メールサーバ106が接続されている。ここで、メールサーバ106を記載しているのは、課金の際に電子メールを使うという意図である。

【0033】

集計サーバ103は、ユーザの使用したサーバリソースを定期的に各サーバから収集し集計して、請求書の発行などの課金処理をおこなう。

【0034】

サーバで使用されたサーバリソースの情報を、集計サーバ103が収集・集計し課金する手順の概略については、後にフローチャートにより説明する。

【0035】

クライアント端末109は、ユーザがサーバ101a、101b、101cにアクセスするために使用され、そのときユーザからの処理要求は、インターネット108、ゲートウェイ107、サービス用ネットワーク110を介して送信される。

【0036】

コンソール端末104は、ユーザがサーバ101a、101b、101cを使用するのに先立ち、システム管理者がユーザの情報を集計サーバ103に登録するために使用される。

【0037】

次に、図2を用いてサーバ101aの内部構成について説明する。

図2は、サーバ101aの内部構成図である。

【0038】

なお、サーバ101b、101cも同様の構成であるため、ここではサーバ101aのみを説明する。

【0039】

サーバ101aは、CPU201a、201bとチップセット202を備え、これらはCPUバス210で接続されている。図2ではサーバ101aは、2個のCPUを搭載しているものとして図示しているが、CPUの個数は2個でなくともシステムの性能要求に応じて適当なCPUの個数を実装することができる。

【0040】

チップセット202には、CPUバス210に加え、主記憶装置203とI/Oバス205が接続されている。主記憶装置203は、CPU201a、201bからチップセットを介してアクセスされる。

【0041】

I/Oバス205には、チップセットに加え、CPUから独立した制御装置で

ある SVP (Service Processor) 206 と、HDD 204 と、NIC (ネットワークインタフェースカード、Network Interface Card) 212 が接続されている。「CPU から独立した」という意味は、SVP 206 は、CPU と独立して処理をおこなうことができ、CPU の動作に影響を受けずに、制御をおこなうことができるという意味である。なお、サーバに CPU とそれとは独立した SVP を設けるのは、従来技術では、通常、CPU を含めたサーバのハードウェア環境を監視するためであることが多い。

【0042】

SVP 206 は、チップセット 202 を経由して、主記憶装置 203 にアクセス可能である。HDD 204 は、CPU 201a、201b からアクセスされる。NIC 212 は、サービス用ネットワーク 110 に接続されており、CPU で動作する OS やアプリケーションプログラムは NIC 212 を経由して通信をおこなう。

【0043】

I/O バス 205 は、例えば、PCI バスなどの周辺回路と通信をおこなうバスである。

【0044】

SVP 206 には、I/O バスに加え、NIC 211 が接続されている。NIC 211 は、また、管理用ネットワーク 102 に接続されており、SVP 206 は、NIC 211、管理用ネットワーク 102 を経由して、集計サーバ 103 と通信をおこなうことができる。

【0045】

次に、図 3 を用いて SVP 206 の構成と動作について説明する。

図 3 は、SVP 206 の構成と OS 304 との通信インタフェースを示す図である。

【0046】

SVP 206 は、RTC (Real Time Clock) 317 と割り込み生成回路 318 を備える。RTC 317 は、数十マイクロ秒から数十ミリ秒の時間間隔で、定期的に割り込み生成回路 318 へコマンドを送信する。そのコマンドを受信した

割り込み生成回路は、割り込み 314 を生成し、CPU に対して割り込みをおこなう。

【0047】

OS 304 は、SVP デバイスドライバ 303 とプロセス制御部 302 を備えている。

【0048】

そして、SVP 206 からの割り込み 314 を契機にして、SVP デバイスドライバ 303 は、API (Application Interface) 呼び出し 313 によってプロセス制御部 302 へ通信をおこなう。

【0049】

ここでは、割り込み 314 によって SVP デバイスドライバ 303 を動作させるものとしたが、割り込み以外に SVP デバイスドライバ 303 の動作を開始させる方法として、SVP 206 から主記憶装置 203 の特定の領域へ動作の開始を指示する識別子を書き込み、SVP デバイスドライバ 303 が、その特定の領域をポーリングして、その識別子の書き込みを発見したら動作を開始する方法や、その特定の領域の代わりに SVP 206 に、レジスタを用意してそのレジスタを SVP デバイスドライバ 303 がポーリングして動作を開始する方法などが考えられる。

【0050】

プロセス制御部 302 は、ユーザプロセス 301a、301b、301c に対してリソース割り付け 311a、311b、311c をおこない、適宜 CPU 時間などのリソースを各プロセスに割り付けている。同時にプロセス制御部 302 は、各プロセスに対して割り付けたリソース量をリソース割付履歴 330 に記録する。プロセス制御部 302 は、SVP デバイスドライバ 303 からの API 呼び出し 313 を契機にして、リソース割付履歴 330 を読み出して、API 応答 312 を SVP デバイスドライバ 303 に返却する。API 応答 312 は、API 呼び出し 313 に対して、リソース情報を返す応答である。この API 応答 312 に関しては、後に詳細に説明する。

【0051】

A P I 応答 3 1 2 を返却後には、リソース割付履歴 3 3 0 を消去する。

【0052】

A P I 応答 3 1 2 を受信した S V P デバイスドライバ 3 0 3 は、A P I 応答 3 1 2 を集計し、S V P 2 0 6 に対しリソース使用状況 3 1 5 を通知する。S V P 2 0 6 は、リソース使用状況 3 1 5 と、現在時刻 3 4 0 を組にして不揮発メモリ 3 2 0 に追記する。リソース使用状況 3 1 5 は、ユーザごとにリソースの使用状況をまとめたものである。

【0053】

このとき、S V P デバイスドライバ 3 0 3 がリソース使用状況 3 1 5 を S V P 2 0 6 に通知する方法として、主記憶装置 2 0 3 の特定の領域にリソース使用状況 3 1 5 を書き込み、S V P 2 0 6 がその特定の領域をポーリングして、書き込みを検出する方法も考えられる。

【0054】

次に、受信回路 3 3 1 は、N I C 2 1 1 を経由して、集計サーバ 1 0 3 から発行された使用リソース情報送信要求 3 3 2 を受信する。受信回路 3 3 1 は、送信回路 3 2 1 に対して送信要求 3 3 3 を発行する。送信回路 3 2 1 は、不揮発メモリ 3 2 0 に格納されたリソース使用状況の情報とサーバ I D 8 0 1 a を組にして使用リソース情報返信 3 1 9 を作成し、N I C 2 1 1 を経由して集計サーバ 1 0 3 に送信する。この使用リソース情報返信 3 1 9 については、後に詳細に説明する。

【0055】

そして、送信回路 3 2 1 は、使用リソース情報返信 3 1 9 の送信完了を確認して、消去信号 3 3 4 をアサートし、不揮発メモリ 3 2 0 に格納されたリソース使用状況の情報を消去する。

【0056】

以上の処理により、ユーザが使用したリソースに関する情報をサーバで詳細に取得できる。また、本情報は後述するユーザに対する課金に使用されるばかりでなく、サーバシステムの性能評価および最適化の指標にも用いることができる。

【0057】

次に、集計サーバ 1 0 3 の構成と動作について説明する。

図 4 は、集計サーバ 1 0 3 の構成図である。

【 0 0 5 8 】

管理者は、ユーザ情報設定 4 1 1 の処理として、コンソール端末 1 0 4 からユーザ登録ウィンドウ（後述）を操作して、ユーザ情報 4 0 1、サーバ割付情報 4 1 0 の設定をおこなう。

【 0 0 5 9 】

ユーザ情報 4 0 1 は、ユーザに関する詳細な情報である。また、サーバ割付情報 4 1 0 とは、ユーザとサーバの関係を示すものであり、サーバ 1 0 1 a、1 0 1 b、1 0 1 c がどのユーザに割り付けられているかの情報を保持する。なお、ユーザ情報 4 0 1、サーバ割付情報 4 1 0 の詳細については、後に説明する。

【 0 0 6 0 】

ユーザ情報設定 4 1 1 の処理についても後にフローチャートにより、詳細に説明する。

【 0 0 6 1 】

また、集計サーバは、定時処理開始指示部 4 2 0 を有する。定時処理開始指示部 4 2 0 は、課金処理を毎日一定の時刻に開始することを可能にするもので、定時処理開始指示部 4 2 0 は、例えば毎日 0 0 時 0 0 分に使用リソース情報送信要求 3 3 2 を管理用ネットワーク 1 0 2 へ送信する。このとき、サーバ割付情報 4 1 0 にサーバ ID 8 0 1 d が登録されているサーバに対してのみ、使用リソース情報送信要求 3 3 2 を送信するようにしてもよい。

【 0 0 6 2 】

サーバ 1 0 1 a、1 0 1 b、1 0 1 c は、集計サーバ 1 0 3 の定時処理開始指示部 4 2 0 からの使用リソース情報送信要求 3 3 2 を受信し、使用リソース情報返信 3 1 9 を返信する。

【 0 0 6 3 】

集計サーバ 1 0 3 は、サーバ 1 0 1 a、1 0 1 b、1 0 1 c から使用リソース情報返信 3 1 9 を受信する。

【 0 0 6 4 】

後に説明するように、使用リソース情報返信 319 は、サーバ ID 801c、収集時刻 802、ローカルユーザ ID 602d、使用 CPU 時間 804、使用メモリ量 805 を含んでいるが、この内で、サーバ ID 801b、ローカルユーザ ID 602a は、グローバルユーザ ID 特定部 421 に入力され、収集時刻 802d、使用 CPU 時間 804d、使用メモリ量 805d は、リソース使用状況集計部 417 に入力される。

【0065】

リソース使用状況集計部 417 は、受信したグローバルユーザ ID 501b に対応するデータをリソース使用状況記録部 418 から読み出し、使用 CPU 時間 804 を使用した CPU 時間の累計に加算し、使用 CPU 時間 804 と使用メモリ量 805 を掛け合わせたデータを累計使用メモリ量として加算して、リソース使用状況記録部 418 を更新する。集計サーバ 103 は、累計使用 CPU 時間 1002、および累計使用メモリ量 1003 に基づいてユーザに課金する金額を決定する。

【0066】

グローバルユーザ ID 特定部 421 は、サーバ割付情報 410 を参照し、受信したサーバ ID 801b、ローカルユーザ ID 602a に対応するグローバルユーザ ID 501b を特定する。

【0067】

また、リソース使用状況集計部 417 は、リソース使用状況記録部 418 のデータを更新し、同時にリソース使用履歴記録部 424 にデータを追記する。リソース使用状況記録部 418 とリソース使用履歴記録部 424 に保持するデータについては、後に詳細に説明する。

【0068】

さらに、定時処理開始指示部 420 は、毎月一回、例えば、末日に請求情報生成要求 423 を発行し、課金情報生成部 419 は、請求情報生成要求 423 を契機に課金処理を実施する

課金処理として請求情報生成部 419 は、ユーザ情報 401、リソース使用状況記録部 418 に記録された使用した CPU 時間の累計と、使用したメモリの累

計の各情報に基づきユーザに課金する金額を決定する。

【0 0 6 9】

決定した請求金額と、リソース使用履歴記録部 4 2 4 に記録された情報とを、請求情報 4 0 7 a、4 0 7 b に記載する。請求情報 4 0 7 a、4 0 7 b は、それぞれプリンタ 1 0 5、メールサーバ 1 0 6 に転送され、プリンタ 1 0 5 では、請求書 4 0 5 として紙面に出力され、メールサーバ 1 0 6 では、電子メール 4 0 6 として電子的にユーザに送信される。これにより管理者は、ユーザの使用したサーバリソースに基づいて、ユーザに課金することが可能になる。

(II) 本実施形態のリソース集計システムの提供するユーザインタフェース

次に、図 5 を用いて本実施形態のリソース集計システムの提供するユーザ登録のためのユーザインタフェースについて説明する。

図 5 は、ユーザ情報設定 4 1 1 のときにコンソール端末に表示される画面の例を示す模式図である。

【0 0 7 0】

ユーザ登録をおこなうときには、ユーザ登録ウィンドウ 1 8 0 1 が、コンソール端末 1 0 4 の C R T などの表示装置に表示される。ユーザ登録ウィンドウ 1 8 0 1 は、M i c r o s o f t 社の W i n d o w s (R) に代表されるグラフィカルユーザインタフェース (G U I) を有する O S のプログラムの一つとして実現される。

【0 0 7 1】

また、S u n M i c r o s y s t e m s 社の J A V A (TM) におけるサーブレットなどのサーバプログラムによって生成された H T M L コンテンツとして実現され、W W W ブラウザによって表示されるようにしてもよいし、他の実現手段を用いてもよい。

【0 0 7 2】

管理者は、1 8 0 2 a ないし 1 8 0 2 e のユーザ情報入力フォームに対してユーザの情報として、ユーザ名 5 0 2 b、グローバルユーザ I D 5 0 1 f、メールアドレス 5 0 4 b、口座番号 5 0 3 b、備考 1 8 1 5 の各情報を入力する。ユーザ名 5 0 2 b は、そのユーザをユーザ自身、および、管理者が識別するための名

前である。

【0073】

グローバルユーザID501fは、集計サーバがそのユーザを識別するための識別子で、本実施形態では、このシステム内で一意の番号を割り付ける。すなわち、グローバルユーザID501fは、本リソース集計システムのために用いられ専用のIDであるという意味合いを持つ。

【0074】

メールアドレス504bは、図4に示したように集計サーバ103がメールサーバ106を介して、そのユーザに電子メールによって請求書を送信するために使用するメールアドレスである。

【0075】

口座番号503bは、そのユーザにサーバの使用料を課金するための課金手段である。なお、課金手段として銀行の口座以外にもクレジットカードを利用して課金し、本フィールドにクレジットカード番号を記録するなどの類似の方法が多数考えられる。

【0076】

備考1815は管理者の利便のため、上記の項目には当てはまらないユーザに関する情報を記録するために使用される。

【0077】

この例では、ユーザ名502bに”B会社”、グローバルユーザID501fに”1001”、メールアドレス504bに”yyy@b-company.com”、口座番号503bに”1230011”、備考1815に”(株)B証券 情報システム部”の文字列がそれぞれ入力されている。なお、このユーザは、法人ユーザとして、サーバリソースを使用する契約を結んでいる場合を想定している。

【0078】

使用サーバ指定ウィンドウ1803は、そのユーザの使用するサーバを指定するためのウィンドウである。管理者は、1806aないし1806cのサーバ割付情報入力フォームに、そのユーザが使用するサーバを指定するサーバID80

1 e、および、そのサーバで使用するローカルユーザ I D 6 0 2 f、ローカルユーザ I D 6 0 2 f に対応するパスワード 1 8 1 8、および、再入力パスワード 1 8 1 9 を入力する。この例では、そのユーザが使用できるサーバを 3 つまで指定できるものとしているが、必ずしも 3 つに制限されるものではない。

サーバ I D 8 0 1 e は、そのシステム内でサーバを一意に特定するための識別子で、本実施例ではシステム内で一意の番号を割り付ける。なお、サーバ I D は該サーバの完全修飾ドメイン名 (Fully Qualified Domain Name, FQDN) のように、システム内で一意な文字列であってもよい。

【 0 0 7 9 】

また、サーバ割付情報とは、後に詳細に説明するが、ユーザとサーバの関係を示す情報である。

【 0 0 8 0 】

ローカルユーザ I D 6 0 2 f は、サーバ I D 8 0 1 e で指定されるサーバ内で、ユーザを一意に特定するための識別子で、本実施形態ではサーバ内で一意の番号を割り付ける。ローカルユーザ I D 6 0 2 f は、OS の管理するユーザ I D を用いることができるが、サーバ内で一意に識別することができれば、他の I D 番号を割り振ってもよい。また番号でなく OS のユーザネームなど一意な文字列であってもよい。

【 0 0 8 1 】

パスワード 1 8 1 8 は、そのユーザがサーバ I D 8 0 1 e で示されるサーバにログインする場合、または、サーバレットなどを通じて処理を依頼する場合に、ユーザの認証をおこなうために必要なパスワードである。サーバにログインする場合には、ユーザは、OS のログイン画面においてローカルユーザ I D 1 8 1 7 とパスワード 1 8 1 8 を入力する。

【 0 0 8 2 】

サーバレットなどを通じて処理を依頼する場合には、クライアント端末 1 0 9 の WWW ブラウザの画面上において、ローカルユーザ I D 1 8 1 7 とパスワード 1 8 1 8 を入力する。

【 0 0 8 3 】

この例ではパスワードの覗き見を防止するため、使用サーバ指定ウィンドウ 1803 中のパスワード 1818 の入力フォームに入力した文字は、全てアスタリスク記号がエコーバックされる。このときパスワードの誤入力を防止するため、管理者は再入力パスワード 1819 にパスワード 1818 で入力したものと同一文字列を入力する。パスワード 1818 と再入力パスワード 1819 の内容が一致していない場合、その旨を表示して管理者に再入力を促す。この例ではサーバ割付情報入力フォーム 1806a に、サーバ ID "0"、ローカルユーザ ID "201" が入力され、サーバ割付情報入力フォーム 1806b にサーバ ID "1"、ローカルユーザ ID "200" が入力されている。これらの情報を入力後、管理者は確認用ボタン 1805a ないし 1805c のいずれかを GUI 操作によって押下する。「はい」の確認用ボタン 1805a が押下された場合、ユーザ情報入力フォーム 1802a ないし 1802e、およびサーバ割付情報入力フォーム 1806a ないし 1806c で入力された各情報が集計サーバ 103 に格納される。また必要に応じて、1806a ないし 1806c で入力された情報が、サーバ ID 801e で指定されたサーバにも格納される。「いいえ」の確認ボタン 1806b、および、「取り消し」の確認ボタン 1806c が押下された場合、入力された各情報は取り消され、集計サーバ 103 にも格納されない。

(III) 本実施形態のリソース集計方法で用いられるデータ構造

次に、図 6 ないし図 12 を用いて本実施形態のリソース集計方法で用いられるデータ構造について説明する。

図 6 は、ユーザ情報 401 のデータ形式を示す模式図である。

図 7 は、API 応答 312 のデータ形式を示す模式図である。

図 8 は、リソース使用状況 315 のデータ形式を示す模式図である。

図 9 は、リソース情報返信 319 のデータ形式を示す模式図である。

図 10 は、サーバ割付情報 410 のデータ形式を示す模式図である。

図 11 は、リソース使用状況記録部 418 に保持されるデータの形式を示す模式図である。

図 12 は、リソース使用履歴記録部 424 に保持されるデータの形式を示す模式図である。

【0084】

図6に示されるユーザ情報401は、図4に示されるように、コンソール端末104から集計サーバ103に対して登録される情報であり、グローバルユーザID501a、ユーザ名502、口座番号503、メールアドレス504の各フィールドを含んでいる。

【0085】

この例では、1行目に、グローバルユーザIDの番号1000で示されるユーザについて、ユーザ名は、“A会社”であり、そのユーザの口座番号は、1230001、メールアドレスが“xxx@acompny.com”であることを示している。

【0086】

図7に示されるAPI応答312は、OS304の中でSVPデバイスドライバのAPI呼び出し313に対して、プロセス制御部302が返す応答であり、OS304内で一意にプロセスを識別するための識別子であるプロセスID601をキーとして、そのプロセスの所有者であるユーザのローカルユーザID602b、そのプロセスが使用した使用CPU時間603と使用メモリ量604の各フィールドが含まれている。

【0087】

この例では、1行目に、プロセスIDの100番で示されるプロセスの所有者はローカルユーザIDの200番で示されるユーザであり、前回のAPI呼び出し313以降に、そのプロセスによって使用されたCPU時間は、10マイクロ秒、使用されたメモリ量は、10MBであることを示している。すなわち、API応答312によって、SVPデバイスドライバ303は、プロセスID単位で使用されたリソース量を把握することができる。

【0088】

図8に示されるリソース使用状況315は、OS304のSVPデバイスドライバ303がSVP206の不揮発メモリ320に書き込む情報であり、ローカルユーザID602cをキーとして、そのユーザが使用した使用CPU時間703、および、使用メモリ量704の各フィールドが含まれる。

【 0 0 8 9 】

この例では 1 行目に、前回の割り込み 3 1 4 の後、ローカルユーザ ID の 2 0 0 番で示されるユーザに所有されるプロセスによって使用された CPU 時間が、3 0 マイクロ秒であり、使用されたメモリ量が 2 0 MBであることを示している。なお、使用 CPU 時間 7 0 3 および使用メモリ量 7 0 4 は、それぞれ API 応答 3 1 2 の使用 CPU 時間 6 0 3 および使用メモリ量 6 0 4 を、ローカルユーザ ID 6 0 2 b が同一である行の数値を足し合わせたものとなっている。すなわちリソース使用状況 3 1 5 によって、SVP 2 0 6 は、ローカルユーザ ID 単位で使用されたリソース量を把握することができる。

【 0 0 9 0 】

図 9 に示される使用リソース情報返信 3 1 9 は、SVP 2 0 6 から集計サーバ 1 0 3 に返信するための情報であり、サーバ ID 8 0 1 c、収集時刻 8 0 2、ローカルユーザ ID 6 0 2 d、使用 CPU 時間 8 0 4、使用メモリ量 8 0 5 の各フィールドを含んでいる。収集時刻 8 0 2 は、前述のリソース使用状況 3 1 5 を不揮発メモリ 3 2 0 に格納した時刻が記録されている。収集時刻 8 0 2 に収集されたリソース使用状況として、対応するローカルユーザ ID 6 0 2 d、使用 CPU 時間 8 0 4、使用メモリ量 8 0 5 が記録されている。

【 0 0 9 1 】

この例では 1 行目に、サーバ ID の番号 0 で示されるサーバにおいて、2 0 0 3 年 0 1 月 1 3 日、1 3 時 0 2 分 1 1 コンマ 0 0 . 0 1 秒に、ローカルユーザ ID の番号 2 0 0 で示されるユーザによって CPU 時間が 3 0 マイクロ秒、メモリ量が 2 0 MB 使用されたことを示している。

【 0 0 9 2 】

図 1 0 に示されるサーバ割付情報 4 1 0 は、図 4 に示されるように、コンソール端末 1 0 4 から集計サーバ 1 0 3 に対して登録される情報であり、サーバ ID 8 0 1 d、ローカルユーザ ID 6 0 2 e、グローバルユーザ ID 5 0 1 c の各フィールドを保持する。

【 0 0 9 3 】

この例では 1 行目に、グローバルユーザ ID の番号 1 0 0 0 で示されるユーザ

が、サーバIDの番号0で示されるサーバの、ローカルユーザIDの番号200を使用することを示している。この例の2行目では、グローバルユーザIDの番号1001で示されるユーザが、サーバIDの番号0のローカルユーザIDの番号201を使用することを示しており、3行目では、同様のグローバルユーザIDの番号1001で示されるユーザが、サーバIDの番号1のローカルユーザIDの番号200を使用することを示している。

【0094】

このサーバ割付情報410により、集計サーバ103は、サーバIDとローカルユーザIDからグローバルユーザIDを特定することができる。

【0095】

図11に示されるリソース使用状況記録部418に保持されるデータは、グローバルユーザID501d、累計使用CPU時間1002、累計使用メモリ量1003の各フィールドを含んでいる。

【0096】

リソース使用状況記録部418に保持されるデータは、図4に示されるように、集計サーバ103のリソース使用状況集計部417から更新されるデータであった。

【0097】

この例では1行目に、グローバルユーザIDが1001で示されるユーザが、今月使用したCPU時間が累計1時間20分であり、使用したメモリ量が累計2.4GB*hourであることを示している。ここで使用したメモリ量の累計の単位として、1GBのメモリを1時間使用した場合に1となる単位として、GB*hourという単位を用いることしたが、もちろん、他の適当な単位も用いることができる。

【0098】

図12のリソース使用履歴記録部424に保持されるデータは、グローバルユーザID501e、日付1202、使用CPU時間1203、使用メモリ量1204の各フィールドを含んでいる。

【0099】

リソース使用履歴記録部 424 に保持されるデータは、図 4 に示されるように、集計サーバ 103 のリソース使用状況集計部 417 から書き込まれるデータデータであった。

【0100】

この例では 1 行目に、グローバルユーザ ID の番号が 1001 で示されるユーザが、2003 年 01 月 01 日に、CPU 時間を 5 分間、メモリを 0.2GB * hour 使用したことを示している。

【0101】

集計サーバ 103 は、本情報を請求書に記載することにより、課金の根拠を明らかにすることができる。

(IV) 本実施形態のリソース集計方法の手順

次に、図 13 および図 14 を用いて本実施形態のリソース集計方法の手順について説明する。

図 13 は、ユーザ情報 401 を集計サーバ 103 に登録するときの、集計サーバにおける処理手順を示すフローチャートである。

【0102】

これは、図 4 に示したユーザ情報設定 411 の処理に相当するものである。

【0103】

集計サーバ 103 は、ユーザの登録を要求されているか判断する（ステップ 1301）。

【0104】

コンソール端末 104 からユーザの登録が要求されていない場合には、本処理を終了する。ユーザの登録が要求されている場合には、ユーザ情報入力受け付けをおこない、必要な情報を入力するよう促す（ステップ 1302）。

【0105】

必要な情報が入力された後に、入力された情報を登録してよいか、管理者に問い合わせる（ステップ 1303）。

【0106】

問い合わせの結果登録可の場合には、入力された情報を集計サーバに格納する

(ステップ1304)。登録不可の場合は、入力された情報を破棄し本処理を終了する。

【0107】

次に、図14を用いてサーバで使用されたサーバリソースの情報を、集計サーバ103が収集・集計し課金する手順の概略について説明する。

【0108】

図14は、サーバで使用されたサーバリソースの情報を、集計サーバ103が収集・集計し課金する手順を示すフローチャートである。

【0109】

この例では、本手順は毎日一定の時刻、例えば、00時00分に開始され、判断1401によって本日が月の末日であるか否かを判断する(ステップ1401)。この判断によって本手順が、毎月末日に行われることが保証される。もちろん、システムや課金の都合により、適当に定められた日におこなうことができる。次に、ユーザが使用したサーバリソース情報の送信を、各サーバのSVP206に要求する(ステップ1402)。

【0110】

各サーバのSVP206は、そのサーバで使用されたサーバリソースの情報を収集しており、本要求に応答してそのサーバで使用されたサーバリソース情報を返信する。次に、集計サーバ303は、各サーバのSVP206から返信された、各サーバで使用されたサーバリソース情報を受信する(ステップ1403)。次に、受信したサーバリソース情報を登録されたユーザごとに集計する(ステップ1404)。最後に、集計されたサーバリソース情報に基づき、課金情報を出力する(ステップ1405)。

【0111】

〔実施形態2〕

次に、図15を用いて本発明に係る第二の実施形態について説明する。

図15は、サーバアプリケーション1101の構成と、OS304bおよびSVP206bとの通信インタフェースを示す模式図である。

【0112】

第一の実施形態は、SVP206は、OS304の中のSVPデバイスドライバ303に割り込み314をかけ、SVPデバイスドライバ303が、OS304の中のプロセス制御部302にAPI呼び出し313を行うことにより、サーバリソース情報を収集するものであった。

【0113】

本実施形態では、SVPがOSの中のSVPデバイスドライバに割り込みをかけるという手順は同じであるが、サーバリソース情報の収集が、サーバアプリケーションでおこなわれるのが異なっている。

【0114】

すなわち、本実施形態では、サーバ101a、101b、101cにおいてSun Microsystems社のJAVA (TM) におけるサーブレットに代表されるサーバアプリケーションを動作させる。

【0115】

図15に示されるように、サーバアプリケーション1101は、サーブレット1103、スレッド実行制御部1107、ユーザID-スレッドID対応表1110、および、プロファイラエージェント1111から構成されている。

【0116】

サーブレット1103は、処理要求1112をサービス用ネットワーク110より受信する。処理要求1112の受信時にサーブレット1103内のユーザ認証部1102において、ユーザの認証をおこなう。処理要求1112は、ユーザIDとパスワードの情報を含んでおり、ユーザ認証部1102は、その情報と、実施形態1で示したようなユーザ登録ウィンドウ1801で入力されたローカルユーザID602fとパスワード1818の組とを照合することで実現する。

【0117】

この処理により、サーブレット1103は、処理要求1112の要求元のユーザを識別することができる。

【0118】

サーブレット1103は、処理内容に応じて、スレッド生成要求1104によりスレッドの生成をスレッド実行制御部1107に要求する。この例でスレッド

は、サーバアプリケーション 1101 内部での実行単位を意味し、各スレッドは、スレッド ID と呼ばれる識別子によって一意に識別される。スレッド実行制御部 1107 はスレッド ID 通知 1105 によって、生成したスレッドのスレッド ID をサブレット 1103 に通知する。また、スレッド実行制御部は、各スレッドに CPU 時間などのサーバリソースを割り付け、サーバリソースの割り付け履歴をリソース割り付け履歴 1106 へスレッド ID 単位で格納する。スレッド ID 通知 1105 を受信したサブレット 1103 は、通知されたスレッド ID と、その処理を要求したユーザのローカルユーザ ID の組を、ユーザ ID - スレッド ID 対応表 1110 に格納する。

【0119】

SVP 206b は、実施形態 1 と同様に、定期的に割り込み 314b を発行する。割り込み 314b を受信した SVP デバイスドライバ 303b は、プロファイラエージェント 1111 に対し、API 呼び出し 313b を実行する。

【0120】

プロファイラエージェント 1111 は、プロファイリング要求 1109 を発行し、スレッド単位リソース使用状況 1108 を受信する。スレッド単位リソース使用状況 1108 には、リソース割付履歴に記録されたスレッド単位でのリソースの割付履歴情報を含んでおり、プロファイラエージェント 1111 はユーザ ID - スレッド ID 対応表 1110 の情報を参照して、ローカルユーザ ID ごとに再集計し、API 応答 312b を OS 304b に送信する。OS 304b は、実施形態 1 の図 3 で説明したようにしてリソース使用状況 315b を SVP 206b に通知し、サーバリソースの情報を収集可能とする。

【0121】

以上によってサーバ 101a、101b、101c でサーバアプリケーションを動作させた場合でも、ユーザの使用したサーバリソースに基づいて、ユーザに課金することができる。

【0122】

〔実施形態 3〕

次に、図 16 ないし図 18 を用いて本発明の第三の実施形態を説明する。

図 1 6 は、使用リソース情報送信要求 3 3 2 b のデータ形式を示す模式図である。

図 1 7 は、使用リソース情報返信 3 1 9 b のデータ形式を示す模式図である。

図 1 8 は、使用リソース情報返信 3 1 9 c のデータ形式を示す模式図である。

【 0 1 2 3 】

第一の実施形態では、集計サーバ 1 0 3 から各サーバに対してなされる使用リソース情報要求 3 3 2 は、特定のユーザを対象としたものではなく、サーバの全ユーザの使用リソース情報を要求するものであった。これに対して、本実施形態は、集計サーバ 1 0 3 から特定のユーザを指定して、使用リソース情報を要求するものである。

【 0 1 2 4 】

すなわち、本実施形態は、集計サーバ 1 0 3 から管理用ネットワーク 1 0 2 に対して、使用リソース情報送信要求 3 3 2 をブロードキャストして、特定のユーザが使用したサーバリソースの情報のみを取得する例を示すものである。

【 0 1 2 5 】

本実施形態では、サーバ 1 0 1 a、1 0 1 b、1 0 1 c で共通のローカルユーザ ID を使用することを前提としている。

【 0 1 2 6 】

集計サーバ 1 0 3 は、ユーザが使用したサーバリソースの情報の送信を各サーバの S V P 2 0 6 に対して要求する際に、図 1 6 に示される使用リソース情報送信要求 3 3 2 b を、管理用ネットワーク 1 0 2 へブロードキャストする。

【 0 1 2 7 】

使用リソース情報送信要求 3 3 2 b には、対象ローカルユーザ ID 1 5 0 2 を含み、これによって収集の対象とする使用リソース情報を指定する。この例では、ローカルユーザ ID が ” 2 0 0 ” であるユーザの使用したサーバリソースの情報のみを各 S V P が送信することを要求している。

【 0 1 2 8 】

S V P 2 0 6 は、対象ローカルユーザ ID 1 5 0 2 で指定されたユーザの使用したサーバリソースの情報を保持している場合、図 1 7 に示される使用リソース

情報返信 319b を返信し、保持していない場合、図 18 に示される使用リソース情報返信 319c を返信する。

【0129】

使用リソース情報返信 319b では、収集時刻 802b、ローカルユーザ ID 602g、使用 CPU 時間 804b、使用メモリ量 805b の各フィールドを含んでいる。なお、本実施形態では、サーバ 101a、101b、101c で共通のローカルユーザ ID を使用することを前提としているため、使用リソース情報返信 319b は使用リソース情報 319 と異なり、サーバ ID の情報を含んでいない。

【0130】

使用リソース情報返信 319c では、データフォーマットは使用リソース情報 319b と同一であるが、使用リソース情報送信要求 332b の対象としたユーザが使用したサーバリソース情報を SVP 206 が所持していないことを示すため、ローカルユーザ ID 602h のフィールドに “-1” という特別な ID を指定する。

【0131】

使用リソース情報返信 319c を受信した集計サーバでは、ローカルユーザ ID 602h に “-1” が指定されていることを認識して、この返信情報を破棄する。

【0132】

集計サーバ 103 は使用リソース情報 319b、および使用リソース情報 319c の受信数をカウントし、全サーバの SVP 206 から返信が返ってきたことを認識し、処理 1404 による受信したリソース情報を登録されたユーザごとに集計する処理を完了する。

【0133】

以上によってユーザの使用したサーバリソースに基づいて、ユーザに課金することが可能になる。

【0134】

〔実施形態 4〕

次に、図 19 を用いて本発明の第四の実施形態を説明する。

図 19 は、論理分割されたサーバにおける SVP 206 c と、OS 304 c、OS 304 d との通信インタフェースを示す模式図である。

【0135】

実施形態 1 と異なり、本実施形態では複数の論理区画 (Logical Partition、以下、LPAR) に分割されたサーバにおいて、使用リソース情報を取得する例を示すものである。

【0136】

複数の論理区画に分割されたサーバでは、サーバリソースの一部分を個々の LPAR に割り付け、LPAR ごとに一つの OS および複数のアプリケーションソフトウェアが稼動する。図 19 に示す例では、サーバは 2 つの論理区画、すなわち LPAR#0 1901 a および LPAR#1 1901 b に分割されている。なお、図 19 では、紙面の都合上 LPAR の数を 2 としたが特に 2 に限定するものではない。

【0137】

LPAR#0 1901 a および LPAR#1 1901 b とは類似の構成であるため、ここでは LPAR#0 1901 a のみを説明する。LPAR 1901 a では、一つの OS 304 c およびアプリケーションソフトウェアに相当するユーザプロセス 301 a ないし 301 c が稼動している。実施形態 1 と同様、SVP デバイスドライバ 303 c は割り込み 314 c を受信し、リソース使用状況 315 c を返信する。

【0138】

SVP 206 c は、本サーバで使用される SVP である。実施形態 1 と異なり、割り込み生成回路 318 b が 2 つの割り込み 314 c および 314 d を有する。割り込み 314 c は LPAR#0 1901 a に対応し、割り込み 314 d は、LPAR#1 1901 b に対応しており、それぞれ SVP デバイスドライバ 303 c および SVP デバイスドライバ 303 d を起動する。

【0139】

実施形態 1 と同様、RTC 317 は数十マイクロ秒から数十ミリ秒の時間間隔

で、定期的に割り込み生成回路 318b へコマンドを送信する。割り込み生成回路 318b はコマンド受信時に、割り込み 314c および割り込み 314d の両方を生成し、CPU に対して割り込みを行う。

【0140】

SVP デバイスドライバ 303c ないし 303d は、リソース使用状況 315c ないし 315d を返信し、それぞれ不揮発メモリ 320b ないし 320c へ、現在時刻 340 と共に格納される。

【0141】

実施形態 1 と同様、受信回路 331 は、NIC 211 を経由して、集計サーバ 103 から発行された使用リソース情報送信要求 332 を受信し、送信要求 333 を発行する。送信回路 321b は、第 1 に LPAR 番号 1910 を 0' にして、LPAR#0 1901a に対応する不揮発メモリ 320b およびサーバ ID 801f を選択し、使用リソース情報返信 319 を送信する。使用リソース情報返信 319 の送信完了を確認後、消去信号 334 をアサートし、不揮発メモリ 320b に格納された LPAR#0 1901a に対応するリソース使用状況の情報を消去する。

【0142】

第 2 に送信回路 321b は、LPAR 番号 1910 を 1' にして、LPAR#1 1901b に対応する不揮発メモリ 320c およびサーバ ID 801g を選択し、使用リソース情報返信 319 を送信する。使用リソース情報返信 319 の送信完了を確認後、消去信号 334b をアサートし、不揮発メモリ 320c に格納された LPAR#1 1901b に対応するリソース使用状況の情報を消去する。

【0143】

以上によって複数の論理区画に分割されたサーバにおいて、使用リソース情報を取得することが可能となる。

【0144】

〔実施形態 5〕

次に、図 20 を用いて本発明の第五の実施形態を説明する。

図20は、論理分割されたサーバにおけるSVP206dと、ハイパバイザ2003、OS304c、OS304dとの通信インタフェースを示す模式図である。

【0145】

実施形態4と異なり、本実施形態ではSVP206dが、唯一つの割り込み314を介して、論理分割されたサーバの使用リソース情報を取得する例を示すものである。

【0146】

ハイパバイザ2003は各LPARへのサーバリソースの割り付けを制御するコンポーネントであり、ファームウェアとして実現される。SVP206dとハイパバイザ2003とは 割り込み314およびリソース使用状況+サーバID2007によって相互にアクセスする。ハイパバイザ2003とLPAR#01901aおよびLPAR#11901bとは、仮想割り込み2005aないし2005b、およびリソース使用状況315cないし315dによって相互にアクセスする。

【0147】

本実施形態では、割り込み314が生成されるとハイパバイザ2003のうち、割り込み転送部2006が起動される。割り込み314を受信した割り込み転送部2006は仮想割り込み2005aおよび2005bを生成し、それぞれSVPデバイスドライバ303cおよび303dを起動する。本処理は、OS304cおよび304dの有する外部割り込みベクタをハイパバイザが参照し、割り込み314に対応する割り込みハンドラを起動することで実現される。

【0148】

仮想割り込み2005aないし2005bを受信したSVPデバイスドライバ303cないし303dは、リソース使用状況315cないし315dをそれぞれ返信する。リソース使用状況315cないし315dは、それぞれ、仮想不揮発メモリ2001aないし2001bに格納される。仮想不揮発メモリ2001aおよび2001bはハイパバイザ用のメモリ領域に確保される。

【0149】

アクセス調停部 2004 は、仮想不揮発メモリ 2001a および 2001b に格納されたりソース使用状況と、サーバ ID 801f ないし 801g を組にして、リソース使用状況+サーバ ID 2007 を SVP 206d に対して送信する。

【0150】

リソース使用状況+サーバ ID 2007 を受信した SVP 206d は、現在時刻 340 とリソース使用状況+サーバ ID 2007 を組にして不揮発メモリ 320d に格納する。

【0151】

送信回路 321 は送信要求 333 受信時に、リソース使用状況 315e およびサーバ ID 801h を不揮発メモリ 320d より読み出し、使用リソース情報返信 319 を送信する。使用リソース情報返信 319 の送信完了を確認後、消去信号 334 をアサートし、不揮発メモリ 320d に格納されたりソース使用状況の情報を消去する。

【0152】

以上によって複数の論理区画に分割されたサーバにおいて、唯一つの割り込みを介して使用リソース情報を取得することが可能となる。

【0153】

【発明の効果】

本発明によれば、サーバの負荷が高い場合であっても、正確にリソースの収集ができ、既存の OS や CPU のアーキテクチャでも変更することなく導入できるサーバリソース集計方法を提供することができる。

【図面の簡単な説明】

【図 1】

本発明の第一の実施形態における計算機システム全体の構成図である。

【図 2】

サーバ 101a の内部構成図である。

【図 3】

SVP 206 の構成と OS 304 との通信インタフェースを示す図である。

【図 4】

集計サーバ 1 0 3 の構成図である。

【図 5】

ユーザ情報設定 4 1 1 のときにコンソール端末に表示される画面の例を示す模式図である。

【図 6】

ユーザ情報 4 0 1 のデータ形式を示す模式図である。

【図 7】

A P I 応答 3 1 2 のデータ形式を示す模式図である。

【図 8】

リソース使用状況 3 1 5 のデータ形式を示す模式図である。

【図 9】

リソース情報返信 3 1 9 のデータ形式を示す模式図である。

【図 1 0】

サーバ割付情報 4 1 0 のデータ形式を示す模式図である。

【図 1 1】

リソース使用状況記録部 4 1 8 に保持されるデータの形式を示す模式図である。

【図 1 2】

リソース使用履歴記録部 4 2 4 に保持されるデータの形式を示す模式図である。

【図 1 3】

ユーザ情報 4 0 1 を集計サーバ 1 0 3 に登録するときの、集計サーバにおける処理手順を示すフローチャートである。

【図 1 4】

サーバで使用されたサーバリソースの情報を、集計サーバ 1 0 3 が収集・集計し課金する手順を示すフローチャートである。

【図 1 5】

サーバアプリケーション 1 1 0 1 の構成と、O S 3 0 4 b および S V P 2 0 6 b との通信インタフェースを示す模式図である。

【図 1 6】

使用リソース情報送信要求 3 3 2 b のデータ形式を示す模式図である。

【図 1 7】

使用リソース情報返信 3 1 9 b のデータ形式を示す模式図である。

【図 1 8】

使用リソース情報返信 3 1 9 c のデータ形式を示す模式図である。

【図 1 9】

複数の論理区画に分割されたサーバにおける S V P 2 0 6 c と O S 3 0 4 c 、
O S 3 0 4 d との通信インタフェース。

【図 2 0】

複数の論理区画に分割されたサーバにおける S V P 2 0 6 c とハイパバイザ 2
0 0 3 、 O S 3 0 4 c 、 O S 3 0 4 d との通信インタフェース。

【符号の説明】

1 0 1 a , 1 0 1 b , 1 0 1 c …サーバ

1 0 2 …管理用ネットワーク

1 0 3 …集計サーバ

1 0 4 …コンソール端末

1 0 5 …プリンタ

1 0 6 …メールサーバ

1 0 7 …ゲートウェイ

1 0 8 …インターネット

1 0 9 …クライアント端末

1 1 0 …サービス用ネットワーク

2 0 1 a , 2 0 1 b …C P U

2 0 2 …チップセット

2 0 3 …主記憶装置

2 0 4 …H D D

2 0 5 …I / O バス

2 0 6 , 2 0 6 b , 2 0 6 c , 2 0 6 d …S V P

2 1 0…CPUバス
2 1 1, 2 1 1 b, 2 1 2…NIC
3 0 1 a, 3 0 1 b, 3 0 1 c, 3 0 1 d, 3 0 1 e, 3 0 1 f…ユーザプロセス
3 0 2, 3 0 2 b, 3 0 2 c…プロセス制御部
3 0 3, 3 0 3 b, 3 0 3 c, 3 0 3 d…SVPデバイスドライバ
3 0 4, 3 0 4 b, 3 0 4 c, 3 0 4 d…OS
3 1 1 a, 3 1 1 b, 3 1 1 c, 3 1 1 d, 3 1 1 e, 3 1 1 f…リソース割り付け
3 1 2, 3 1 2 b, 3 1 2 c, 3 1 2 d…API応答
3 1 3, 3 1 3 b, 3 1 3 c, 3 1 3 d…API呼び出し
3 1 4, 3 1 4 b, 3 1 4 c, 3 1 4 d…割り込み
3 1 5, 3 1 5 b, 3 1 5 c, 3 1 5 d, 3 1 5 e…リソース使用状況
3 1 7…RTC
3 1 8, 3 1 8 b…割り込み生成回路
3 1 9…使用リソース情報返信
3 1 9 b…使用リソース情報返信
3 1 9 c…使用リソース情報返信
3 2 0, 3 2 0 b, 3 2 0 c, 3 2 0 d…不揮発メモリ
3 2 1, 3 2 1 b…送信回路
3 3 0, 3 3 0 b, 3 3 0 c…リソース割付履歴
3 3 1…受信回路
3 3 2…使用リソース情報送信要求
3 3 2 b…使用リソース情報送信要求
3 3 3…送信要求
3 3 4…消去信号
3 4 0…現在時刻
4 0 1…ユーザ情報
4 0 5…請求書

4 0 6 …電子メール
4 0 7 a, 4 0 7 b …請求情報
4 1 0 …サーバ割付情報
4 1 1 …ユーザ情報設定
4 1 7 …リソース使用状況集計部
4 1 8 …リソース使用状況記録部
4 1 9 …請求情報生成部
4 2 0 …定時処理開始部
4 2 1 …グローバルユーザ I D 特定部
4 2 3 …請求情報生成要求
4 2 4 …リソース使用履歴記録部
5 0 1 a, 5 0 1 b, 5 0 1 c, 5 0 1 d, 5 0 1 e, 5 0 1 f …グローバルユーザ I D
5 0 2, 5 0 2 b …ユーザ名
5 0 3, 5 0 3 b …口座番号
5 0 4, 5 0 4 b …メールアドレス
6 0 1 …プロセス I D
6 0 2 a, 6 0 2 b, 6 0 2 c, 6 0 2 d, 6 0 2 e, 6 0 2 f, 6 0 2 g, 6 0 2 h …ローカルユーザ I D
6 0 3 …使用 C P U 時間
6 0 4 …使用メモリ量
7 0 3 …使用 C P U 時間
7 0 4 …使用メモリ量
8 0 1 a, 8 0 1 b, 8 0 1 c, 8 0 1 d, 8 0 1 e, 8 0 1 f, 8 0 1 g, 8 0 1 h …サーバ I D
8 0 2, 8 0 2 d, 8 0 2 b, 8 0 2 c …収集時刻
8 0 4, 8 0 4 b, 8 0 4 c, 8 0 4 d …使用 C P U 時間
8 0 5, 8 0 5 b, 8 0 5 c, 8 0 5 d …使用メモリ量
1 0 0 2 …累計使用 C P U 時間

1 0 0 3 …累計使用メモリ量
1 1 0 1 …サーバアプリケーション
1 1 0 2 …ユーザ認証部
1 1 0 3 …サーブレット
1 1 0 4 …スレッド生成要求
1 1 0 5 …スレッド I D 通知
1 1 0 6 …リソース割付履歴
1 1 0 7 …スレッド実行制御部
1 1 0 8 …スレッド単位リソース使用状況
1 1 0 9 …プロファイリング要求
1 1 1 0 …ユーザ I D - スレッド I D 対応表
1 1 1 1 …プロファイラエージェント
1 1 1 2 …処理要求
1 1 1 3 …処理結果
1 2 0 2 …日付
1 2 0 3 …使用 C P U 時間
1 2 0 4 …使用メモリ量
1 3 0 1 …ユーザの登録要求有無判断
1 3 0 2 …ユーザ情報の入力受け付け処理
1 3 0 3 …入力情報の登録可否判断
1 3 0 4 …ユーザ情報登録
1 4 0 1 …月の末日判断
1 4 0 2 …サーバリソース情報の送信要求処理
1 4 0 3 …サーバリソース情報の受信処理
1 4 0 4 …受信したリソース情報の集計処理
1 4 0 5 …課金情報出力処理
1 5 0 2 …対象ローカルユーザ I D
1 8 0 1 …ユーザ登録ウィンドウ
1 8 0 2 a, 1 8 0 2 b, 1 8 0 2 c, 1 8 0 2 d, 1 8 0 2 e …ユーザ情報入

カフォーム

1 8 0 3 …使用サーバ指定ウィンドウ

1 8 0 5 a, 1 8 0 5 b, 1 8 0 5 c …確認用ボタン

1 8 0 6 a, 1 8 0 6 b, 1 8 0 6 c …サーバ割付情報入力フォーム

1 8 1 5 …備考

1 8 1 8 …パスワード

1 8 1 9 …再入力パスワード

1 9 0 1 a …L P A R # 0

1 9 0 1 b …L P A R # 1

1 9 1 0 …L P A R 番号

2 0 0 1 a, 2 0 0 1 b …仮想不揮発メモリ

2 0 0 3 …ハイパバイザ

2 0 0 4 …アクセス調停部

2 0 0 5 a, 2 0 0 5 b …仮想割り込み

2 0 0 6 …割り込み転送部

2 0 0 7 …リソース使用状況+サーバ I D。

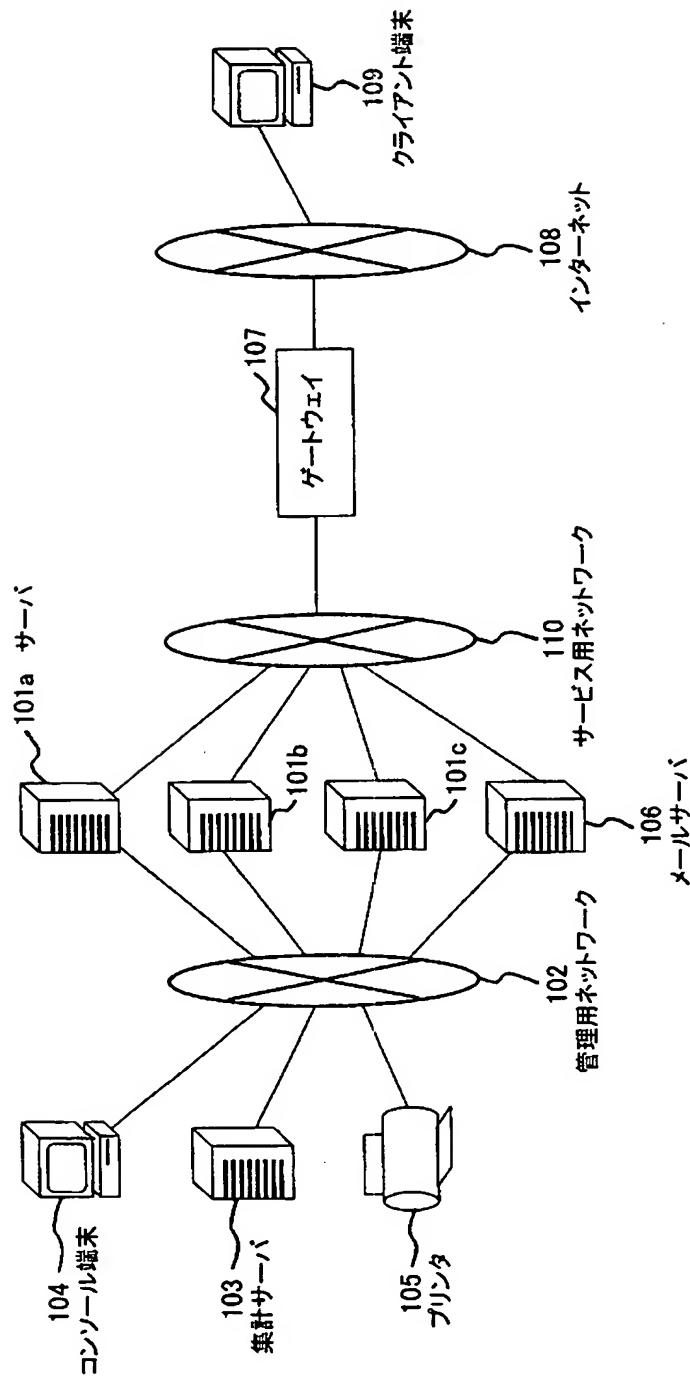
【書類名】

図面

【図 1】

図 1

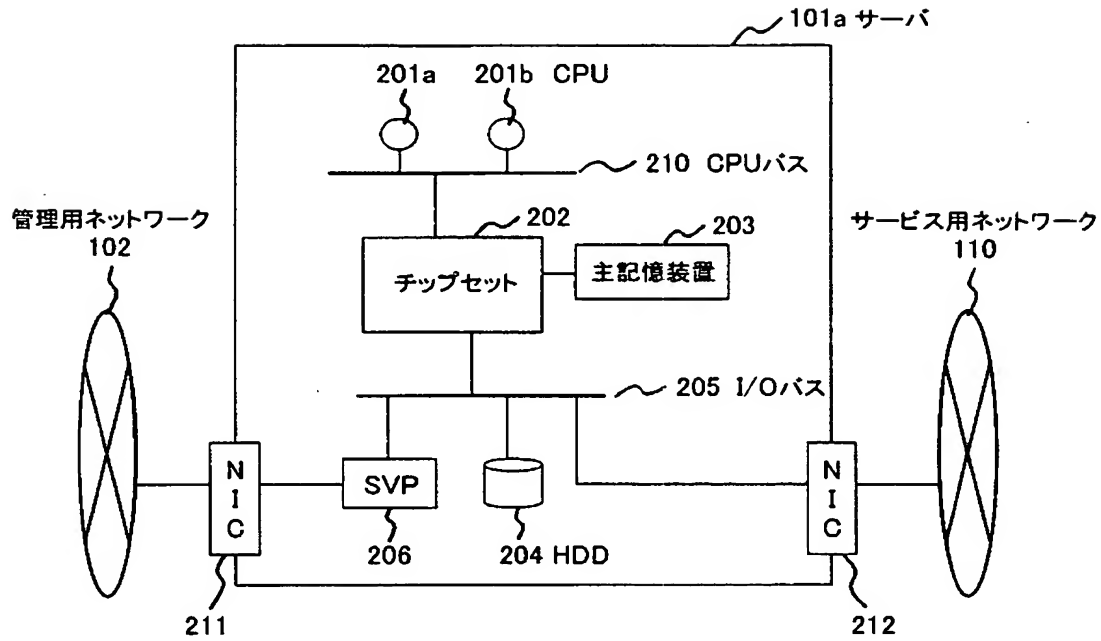
第一の実施形態における計算機システムの構成



【図 2】

図 2

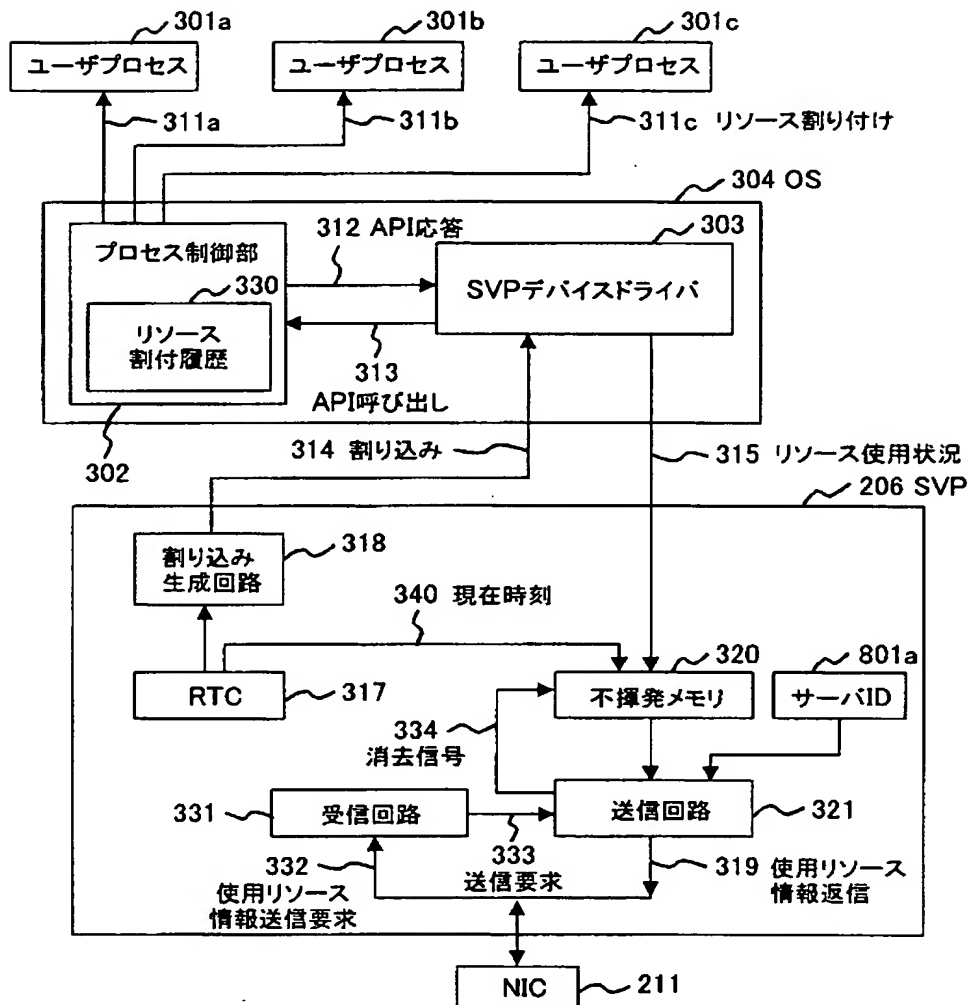
サーバ101aの内部構成



【図 3】

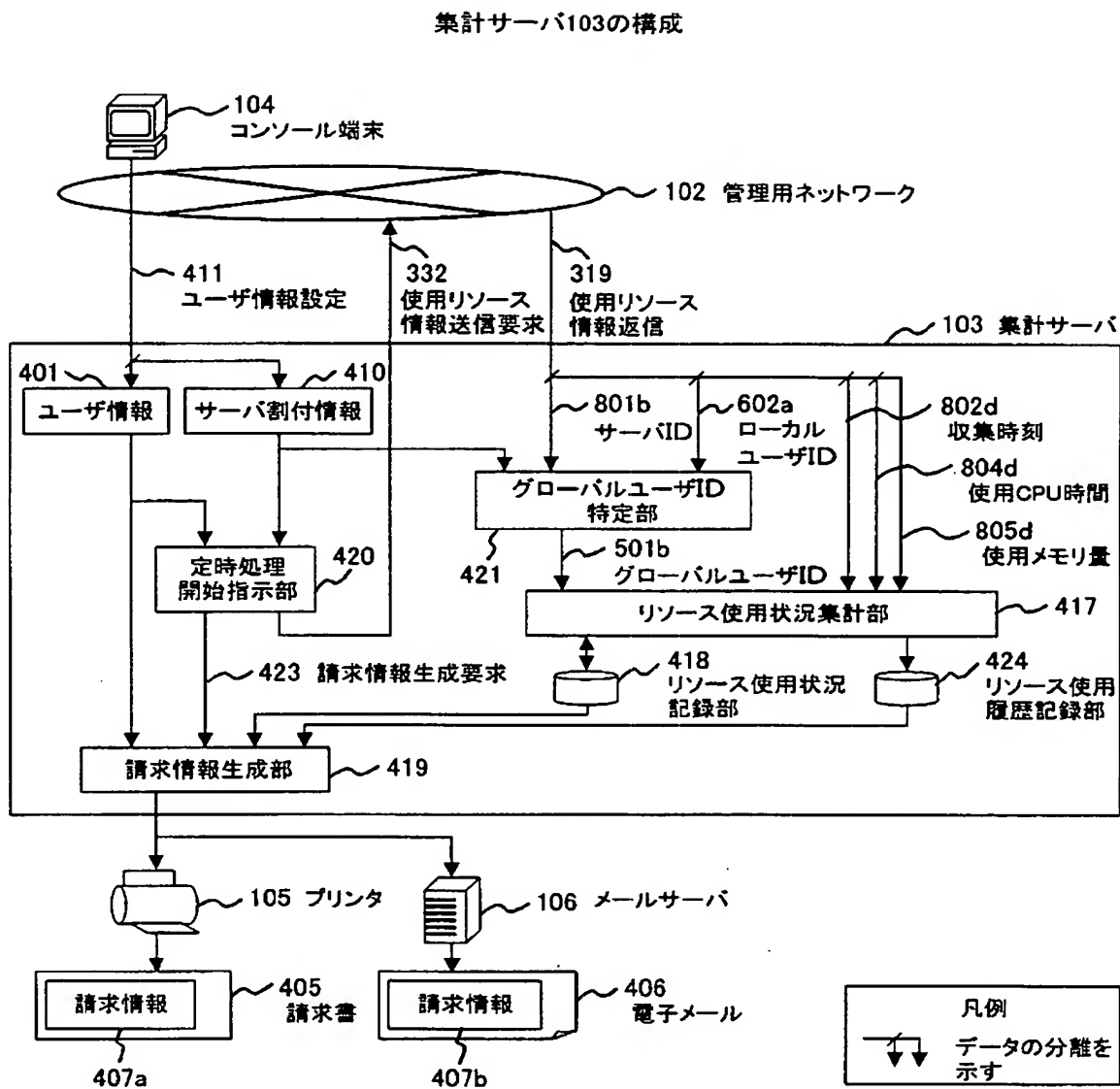
図 3

SVP206の内部構成およびOS304との通信インタフェース



【図 4】

図 4



【図 5】

図 5

ユーザ情報入力受け付け処理1302において、コンソール端末に表示される画面の例

ユーザ情報編集 (新規登録)

502b ユーザ名	B会社	1802a ユーザ情報入力
501f グローバルユーザID	1001	1802b フォーム
504b メールアドレス	yyy@b-company.com	1802c
503b 口座番号	1230011	1802d
1815 備考	(株)B証券 情報システム部	1802e

使用するサーバの指定

サーバID	ローカルユーザID	パスワード	再入力パスワード
801e	602f	1818	1819
0	201	*****	*****
1	200	*****	*****

1803 使用サーバ指定ウィンドウ

1806a 1806b 1806c サーバ割付情報入力フォーム

1805a はい 1805b いいえ 1805c 確認用ボタン

【図 6】

図 6

ユーザ情報401のデータ形式

401 ユーザ情報

501a グローバルユーザID	502 ユーザ名	503 口座番号	504 メールアドレス
1000	A会社	1230001	xxx@a-company.com
1001	B会社	1230011	yyy@b-company.com
1002	C団体	3400123	zzz@c-organization.org

【図 7】

図 7

API応答312のデータ形式

312 API応答

601 }	602b }	603 }	604 }
プロセスID	ローカルユーザID	使用CPU時間	使用メモリ量
100	200	10 μ s	10MB
101	200	20 μ s	10MB
102	201	20 μ s	20MB

【図 8】

図 8

リソース使用状況315のデータ形式

315 リソース使用状況

602c }	703 }	704 }
ローカルユーザID	使用CPU時間	使用メモリ量
200	30 μ s	20MB
201	20 μ s	20MB

【図 9】

図 9

リソース情報返信319のデータ形式

319 使用リソース情報返信

801c サーバID	802 収集時刻	602d ローカルユーザID	804 使用CPU時間	805 使用メモリ量
0	2003/01/13 13:02:11.00.01	200	30 μ s	20MB
0	2003/01/13 13:02:11.00.01	201	20 μ s	20MB
0	2003/01/13 13:02:11.00.00	200	25 μ s	20MB
0	2003/01/13 13:02:11.00.00	201	40 μ s	15MB
⋮	⋮	⋮	⋮	⋮
0	2003/01/13 13:01:00.00.20	200	35 μ s	15MB

【図 10】

図 10

サーバ割付情報410のデータ形式

410 サーバ割付情報

801d サーバID	602e ローカルユーザID	501c グローバルユーザID
0	200	1000
0	201	1001
1	200	1001
2	200	1002

【図 11】

図 11

リソース使用状況記録部418に保持されるデータの形式

418 リソース使用状況記録部

501d グローバルユーザID	1002 累計使用CPU時間	1003 累計使用メモリ量
1001	1h20m	2.40GB*hour
1002	1h10m	2.20GB*hour
1003	2h00m	4.00GB*hour

【図 12】

図 12

リソース使用履歴記録部424に保持されるデータの形式

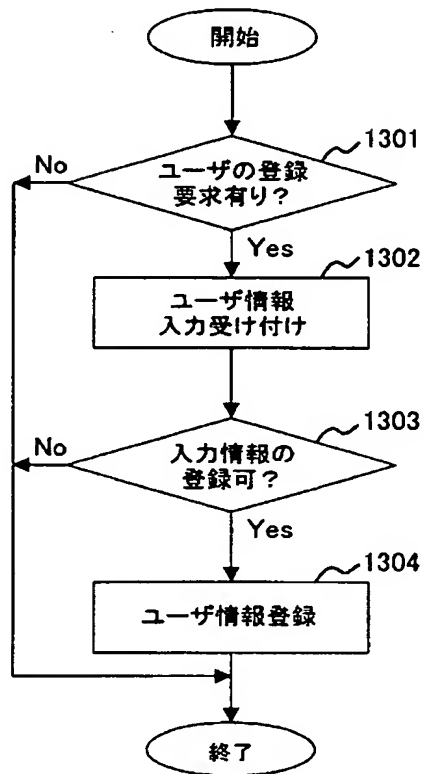
424 リソース使用履歴記録部

501e グローバルユーザID	1202 日付	1203 使用CPU時間	1204 使用メモリ量
1001	2003/01/01	0h05m	0.2GB*hour
1001	2003/01/02	0h05m	0.2GB*hour
1001	2003/01/03	0h10m	0.4GB*hour
⋮	⋮	⋮	⋮
1001	2003/01/31	0h30m	1.2GB*hour
1002	2003/01/01	0h10m	0.5GB*hour
1002	2003/01/02	0h10m	0.5GB*hour

【図 1 3】

図 1 3

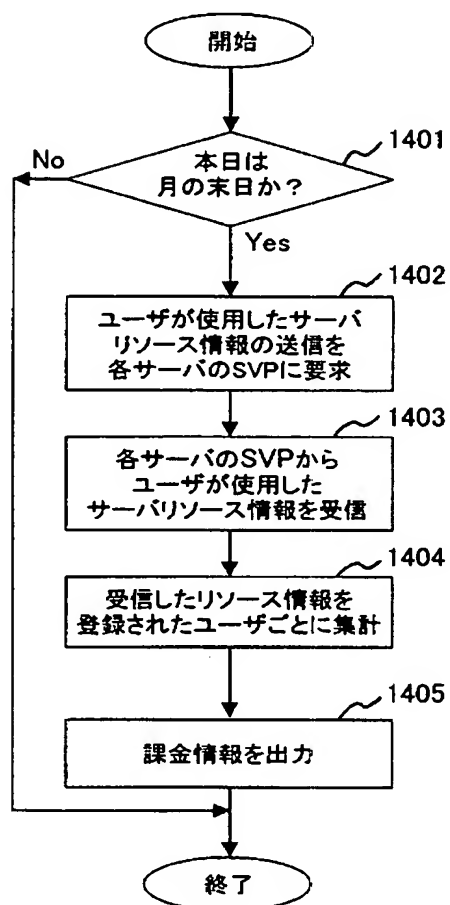
ユーザを集計サーバ103に登録するときの、集計サーバにおける処理手順



【図 14】

図 14

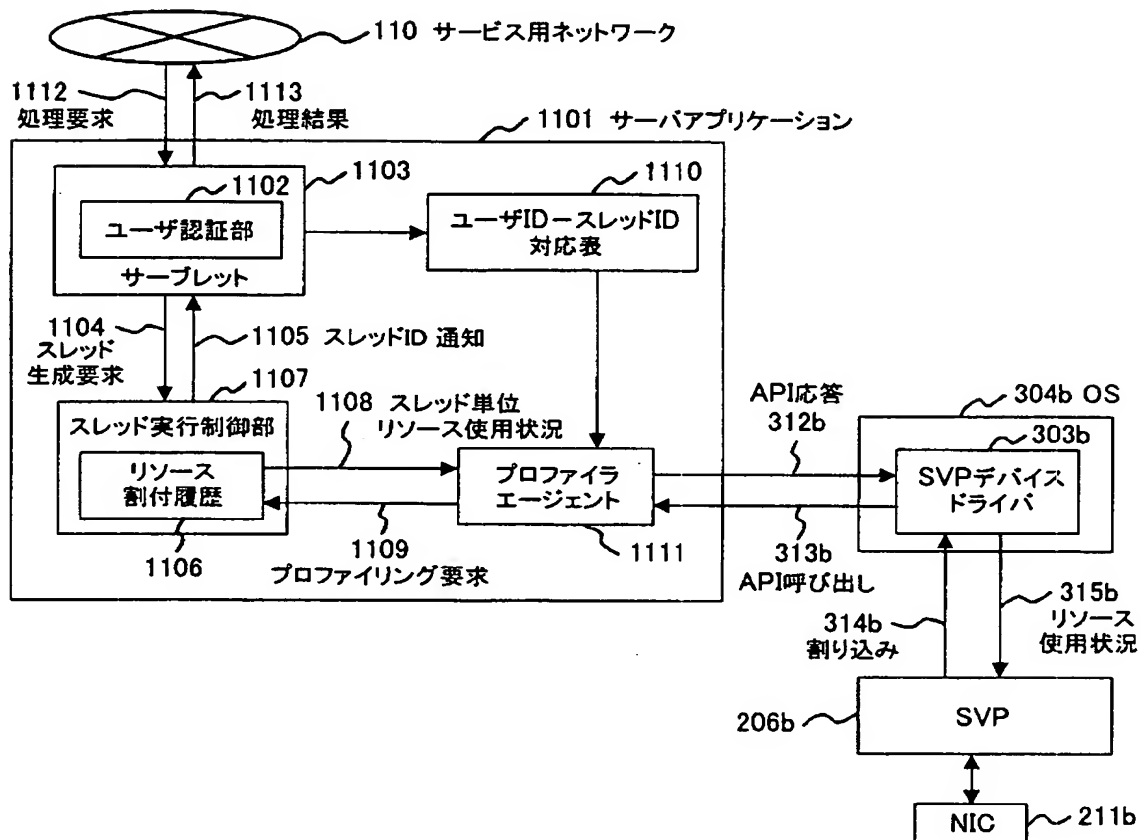
サーバで使用されたサーバリソースの情報を、集計サーバ103が収集する手順



【図 15】

図 15

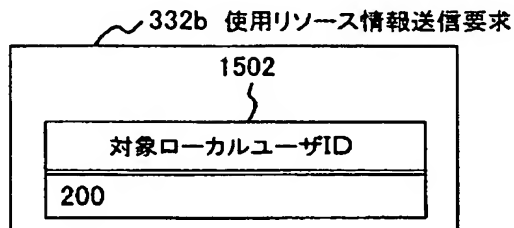
サーバアプリケーション1101の構成と、OS304bおよびSVP206bとの通信インタフェース



【図 16】

図 16

使用リソース情報送信要求332bのデータ形式



【図 17】

図 17

使用リソース情報返信 319b のデータ形式

319b 使用リソース情報返信

802b }	602g }	804b }	805b }
収集時刻	ローカルユーザID	使用CPU時間	使用メモリ量
2003/01/13 13:02:11.00.01	1000	30 μ s	20MB
2003/01/13 13:02:11.00.00	1000	25 μ s	20MB
⋮	⋮	⋮	⋮
2003/01/13 13:01:00.00.20	1000	35 μ s	15MB

【図 18】

図 18

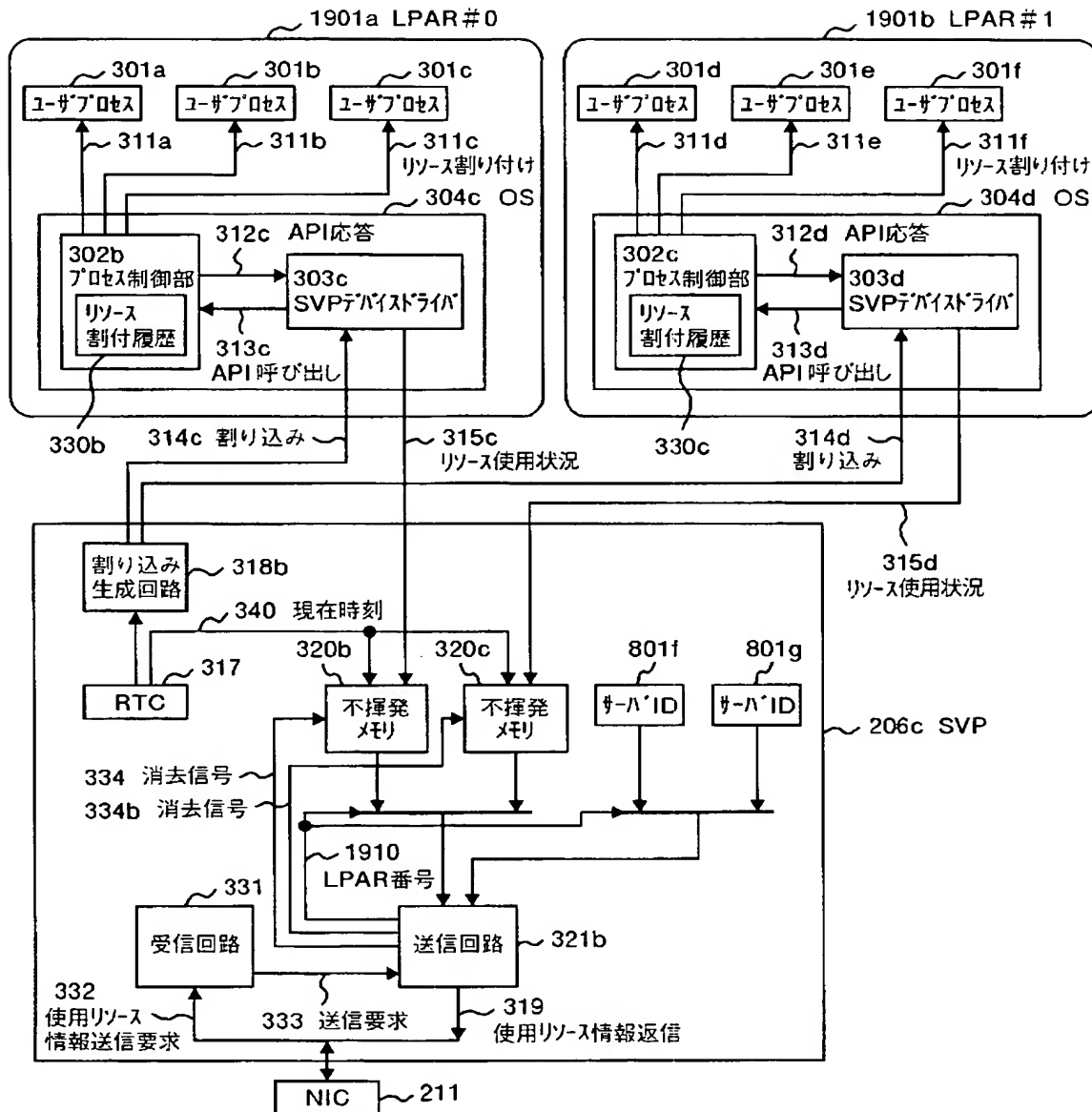
使用リソース情報返信 319c のデータ形式

319c 使用リソース情報返信

802c }	602h }	804c }	805c }
収集時刻	ローカルユーザID	使用CPU時間	使用メモリ量
2003/01/01 00:00:00.00.00	-1	0 μ s	0MB

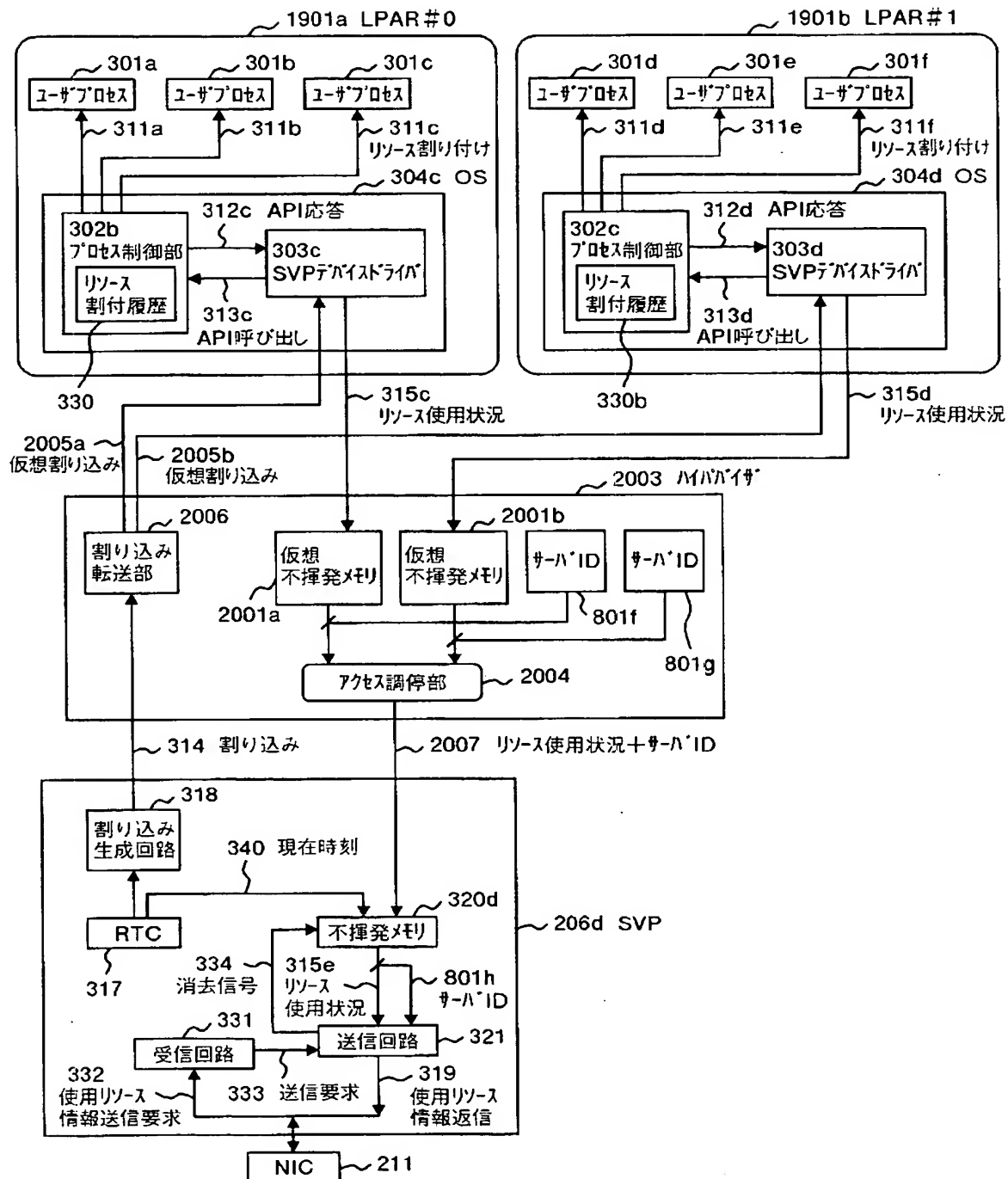
【図 19】

図 19



【図 20】

図 20



【書類名】 要約書

【要約】

【課題】 サーバの負荷によらず正確にサーバリソースの使用状況を把握することができ、しかも、OSやCPUの種類に依存せずに、サーバリソースの集計とそれによる課金をおこなえるようにする。

【解決手段】 サーバにCPUから独立した処理をおこなえる制御装置であるSVP 2 0 6 を設け、CPUに対して割り込み3 1 4 を定期的に実施する。SVP デバイスドライバ3 0 3 は、割り込み3 1 4 を契機にOSが提供するAPIを用いて使用されたCPU時間やメモリ容量などのサーバリソースの数量と使用したユーザの情報を取得しSVP 2 0 6 に渡す。SVP 2 0 6 は、渡されたリソース情報をNIC 2 1 1 を経由して集計サーバに送信する。集計サーバは使用されたサーバリソース量をユーザごとに集計し、使用されたサーバリソースの多寡に応じてユーザに課金する。

【選択図】 図 3

特願 2003-193335

出願人履歴情報

識別番号

[000005108]

1. 変更年月日

1990年 8月31日

[変更理由]

新規登録

住 所

東京都千代田区神田駿河台4丁目6番地

氏 名

株式会社日立製作所